Backstepping Sliding Mode Control of Uncertainty Flexible Joint Manipulator with Actuator Saturation

Hongyu Chen, Xiucheng Dong*, Yong Yang and Jiutai Liu

School of Electrical Engineering and Electronic Information, Xihua University, Chengdu 610039, P. R. China

*Email: dxc136@163.com

Abstract. To address the issue of the flexible joint (FJ) manipulator control with actuator saturation, singular perturbation method is adopted in this paper which decompose the manipulator model into two subsystems in terms of time scale transformation, the fast one represents flexibility and the slow one expresses rigidity, and the control law of which is designed respectively. For the fast subsystem, a velocity-difference-based feedback control is designed to lower the oscillation caused by joint flexibility, while for the slow subsystem, the control performance of the system is enhanced by a combination of a class of nonlinear integral sliding surface and backstepping global sliding mode. The radial basis function (RBF) neural networks are utilized to compensate the actuator saturation, the disturbance and the modeled uncertainties. Based on the Lyapunov theory, the tracking convergence of the closed-loop system is proved rigorously. The simulation result shows that the designed control law can maintain good rapidity and accuracy, moreover, chattering is eliminated.

1. Introduction

Compared with the traditional rigid robot manipulator, the robot manipulator with FJ have the characteristics of less energy consumption, larger load ratio, and smaller contact impact. However, joint flexibility creates obstacles such as hysteresis, nonlinear coupling, etc. It really poses a challenge for the control of the FJ manipulator as to make sure that manipulator maintain high precision, fast speed, and vibration-free performance while preserving its flexibility.

At present, scholars have made much progress in this area by doing a great deal of research on the control of flexible systems. Sliding mode method can enhance the robustness of system, meanwhile, the backstepping method can remove the matching conditions needed in the design of sliding mode control. In [1], neural network adaptive and dynamic surface method are combined to control the flexible joint manipulator with uncertain model. In [2], an adaptive second order terminal sliding mode controller is proposed for controlling robotic manipulators. In [3], a robust repetitive learning scheme is presented by using the backstepping technique for the unidirectional servo valve, aiming to track the movement. [4] proposes an optimal second order integral sliding mode control for a single link FJ manipulator to achieve robust and smooth performance. [5] presents a hierarchical sliding mode control for a rotary FJ manipulator. In the case of motor position information only, a robust sliding mode observer is designed to measure the position and speed information of the linkage accurately in [6]. [7] proposed robust controller based on iterative learning observer. Based on the two type fuzzy neural network, a method combining with adaptive backstepping control is proposed in [8]. In [9], new switch function based on backstepping method is designed to transfer discontinuity to
derivativeness of the control law, so that the chattering of control can be eliminated. In [10], to overcome the impact of unknown and bounded uncertainty, a backstepping RBF neural network sliding mode controller is designed. [11] proposed a method which consists of the repetitive learning control, neural network control and a combined error factor to improve the robustness of the system. In [12], a neural networks based dynamic integral sliding mode control with output differentiator observer is developed.

However, the ideal control law can not be realized in practice, because the actuator often has the characteristics of saturation. To solve this problem, in [13], a periodic adaptive learning algorithm is designed to estimate the auxiliary parameter for approximating and compensating the section which exceeds the saturated limit by a compensator. In [14], to cope with fast time-varying external disturbances, a high order disturbance observer is adopted, moreover, a smooth hyperbolic function is included in the controller to satisfy the requirement of input saturation. In [15], the input saturation is handled by designing an auxiliary system.

Considering the torque saturation of FJ actuator, this paper proposes a new backstepping global sliding mode method to study the trajectory tracking issue. Firstly, based on the principle of singular perturbation, the model of the manipulator with flexible joints is divided into two subsystems according to different time scales: the fast subsystem and the slow subsystem. The former represents the flexibility of the manipulator, the later represents its rigidity. Secondly, the controllers of the two subsystems should be designed separately. For the fast subsystem, the velocity-difference-based feedback method is adopted to suppress the flexible vibration in the manipulator. For the slow system, the method of global sliding mode backstepping is proposed, and a nonlinear integral surface in the first step of backstepping is added to guarantee a better tracking performance. Regarding the saturation of actuator and the inaccuracy of the system, the RBF neural network adaptive method is used to compensate the unexecuted torque and the stability of the controllers is analyzed. Finally, we simulated the controller designed in this paper with MATLAB. The results show the accuracy of the controller and the effectiveness in compensating the actuator saturation.

2. System Modelling and Preliminaries

2.1. System Modelling

Considering the uncertainties of system, the dynamics of n-joint FJ manipulator can be written as:

\[
\begin{align*}
[M_m(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + \chi_1 + \rho_1 = K(\theta - q)] \\
[J(\ddot{q}) + K(\theta - q) + \chi_2 + \rho_2 = \tau]
\end{align*}
\]

where, \( M_m(q) \in \mathbb{R}^{n \times n} \) is the positive definite link inertia matrix, \( C(q, \dot{q}) \in \mathbb{R}^{n \times n} \) denotes the Coriolis and centripetal forces matrix, \( G(q) \in \mathbb{R}^n \) is the gravity vector. \( J \in \mathbb{R}^{n \times n} \) is the positive definite motor inertia matrix, \( K \in \mathbb{R}^{n \times n} \) is a diagonal and positive definite matrix of joint spring constants. \( q, \dot{q}, \ddot{q} \) denote the link position, velocity, and acceleration vectors, respectively. \( \theta, \dot{\theta}, \ddot{\theta} \) denote the actuator position, velocity, and acceleration vectors, respectively. \( \chi_1, \chi_2, \rho_1, \rho_2 \) are the modeling error of connecting link side and motor side, respectively. \( \tau_1, \tau_2 \) are the external disturbance of connecting link side and motor side respectively.

2.2. RBF Neural Network

Any continuous unknown function \( h(X) \) can be approximated as:

\[
h(X) = W^T \phi(x) + e
\]
where \( W^* \in \mathbb{R}^{n \times n} \) known as the ideal constant weights matrix, \( \phi(x) \) \( n \) is the basis function of neural network, \( \epsilon \) is the bounded approximation error vector which satisfies \( \| \epsilon \| \leq \epsilon_N \). In this article, choose \( \phi(x) \) as the Gauss basis function, whose expression is:

\[
\varphi(x) = \exp \left[ -\frac{\| x - c_j \|^2}{2b_j^2} \right]
\]

where \( x \) is the input of neural network. \( c_j \) is the coordinate value of the center vector of the \( j \)th Gaussian basis function; \( b > 0 \) is the width of the Gauss basis function.

### 2.3. Actuator Saturation

When the actuator of the system is saturated, define \( u \) as the input of actuator, \( u_{\text{max}} \) and \( u_{\text{min}} \) is the upper and lower bounds of the actuator output, then the actuator output \( u_i \) can be expressed as:

\[
\tau_i = \text{sat}(u_i) = \begin{cases} 
\tau_{\text{max}}, & u_i > \tau_{\text{max}} \\
\tau_i, & \tau_{\text{min}} \leq u_i \leq \tau_{\text{max}} \\
\tau_{\text{min}}, & u_i < \tau_{\text{min}} 
\end{cases}
\]

(4)

Considering the control signal exceeds the upper and lower saturation limits of the actuator, saturation will occur. The control input \( u \) cannot be fully executed at this time. Define the unimplemented part as \( D \), then:

\[
D_i = u_i - \text{sat}(u_i) = \begin{cases} 
\tau_{\text{max}} - u_i, & u_i > \tau_{\text{max}} \\
0, & \tau_{\text{min}} < u_i < \tau_{\text{max}} \\
\tau_{\text{min}} - u_i, & u_i < \tau_{\text{min}} 
\end{cases}
\]

(5)

### 3. Controller Design and Stability Analysis

#### 3.1. Singular Perturbation Method and Fast Subsystem Controller Design

Singular perturbation has the advantage of reducing the order of the system, which is often used in FJ manipulator control. In this paper, the FJ manipulator is decomposed into two subsystems by singular perturbation method, and the controller of the them is designed respectively.

\[
t = s + f
\]

(6)

The control law of the fast subsystem provides rapid responses to suppress elastic vibration in sudden changes, while controller of the slow is specifically designed for the quasi-steady-state system, and the tracking is realized by compensating the total disturbance of the system.

Define the elastic torque:

\[
z = K(q - q)
\]

(7)

According to the dynamic model of the system, the equation of elastic torque can be described as:

\[
JK^{-1}z + z = \tau_s + \tau_f - J\dot{q} - \chi_i - \rho_i
\]

(8)

Lead into a parameter \( \epsilon \), let \( K = K_\epsilon / \delta^3 \), where \( K_\epsilon \) is a positive definite diagonal matrix. The smaller \( \epsilon \) is, the bigger the joint stiffness is. The fast control law is selected as the speed difference feedback method. In [16], it has been proved that the velocity-difference-based feedback feedback method can
guarantee the global asymptotic stability of the fast subsystem. Therefore, this paper only needs to prove the stability of the slow subsystem:

$$\tau_f = -K_f (\dot{\theta} - \dot{q})$$  \hspace{1cm} (9)

Choose the above formula (9) to get:

$$e^2 J \ddot{e} + c_k K_k \dot{e} = K_e (\tau_s - J \ddot{q} - \chi)$$  \hspace{1cm} (10)

The quasi steady state equation of the system considering the modeling error and external disturbance is obtained:

$$M(q) \ddot{q} + C(q, \dot{q}) \dot{q} + G(q) + \chi + \rho = \tau_s$$  \hspace{1cm} (11)

where:

$$M(q) = M_m(q) + J, \quad c = c_1 + c_2, \quad r = r_1 + r_2$$

3.2. Design and Stability Analysis of Slow Subsystem Controller

A controller that combines the sliding mode control with the backstepping method is designed. To pursue better performance, a nonlinear saturation function that could amplify small errors is proposed. The expected error performance can be tracked easily by selecting different \( b \):

$$g(e) = \begin{cases} \sin \frac{e}{2} & |e| < b \\ e & e \geq b \\ e & e \leq -b \end{cases}$$  \hspace{1cm} (12)

Step 1: define the tracking error and nonlinear integral sliding surface of the system:

$$\begin{cases} e = x_1 - x_{1d} \\ s_1 = e + \int g(e) dt \end{cases}$$  \hspace{1cm} (13)

where \( \lambda > 0 \) is a constant, \( x_{1d} \) is a reference trajectory with second derivative.

$$\begin{cases} \dot{e} = x_2 - \dot{x}_{1d} \\ \dot{s}_1 = \dot{e} + \lambda g(e) = x_2 - \dot{x}_{1d} + \lambda g(e) \end{cases}$$  \hspace{1cm} (14)

Define Lyapunov function \( V_1 = \frac{1}{2} s_1^T s_1 \), its first derivative is:

$$\dot{V}_1 = s_1 \dot{x}_2 - \ddot{x}_{1d} + \lambda g(e) = s_1 (s_2 + \alpha_1 - \dot{x}_{1d} + \lambda g(e))$$  \hspace{1cm} (15)

Step 2: define error variable \( s_2 \) and virtual variable \( \dot{s}_1 \) as

$$s_2 = x_2 - \alpha_1, \dot{s}_1 = \dot{x}_{1d} - \lambda g(e) - k_1 s_1$$  \hspace{1cm} (16)

Further, substituting (16) into (15) gives:

$$\dot{V}_1 = s_1 s_2 - k_1^2 s_1^2$$  \hspace{1cm} (17)

The principle of a global sliding mode could make the system fall into the sliding mode motion from the beginning, so that the whole response process of the system possesses robustness.

The global linear sliding surface is designed as follows:
\( s = c_i s_1 + s_2 + p(t)[c_i s_i(0) + s_2(0)] \)  
\( p(t) = \text{diag}(\exp(-\varphi_1 t), \exp(-\varphi_2 t), \ldots, \exp(-\varphi_n t)), p(t) > 0 \)  

Considering the output torque of the actuator is bounded, the definition of the unexecuted part of the torque, \( u \) is the output of the controller, we can get:

\[
\dot{s}_2 = \dot{x}_2 - \dot{\alpha}_i = M(q)^{-1}(\tau_x - C(q, \dot{q})\dot{q} - G(q) - \chi - \rho) - \dot{\alpha}_i
\]

From (5),

\[
\dot{s} = u + f
\]

Subsequently, combination of (20), and (19) leads to

\[
\dot{s}_2 = M(q)^{-1}(\Delta + u + \tau_f - C(q, \dot{q})\dot{q} - G(q) - \chi - \rho) - \dot{\alpha}_i
\]

Then, we can design the control law \( u_i \) as:

\[
u_i = C(q, \dot{q})\dot{q} + G(q) + \dot{\varphi}(t)[c_i \dot{s}_i(0) + \dot{s}_2(0)] - \tau_f + M(q)[\dot{\alpha} - \dot{c}_i \dot{s}_1 - \frac{s}{\|s\|^2}(s_1^T s_2)] + u_{sw1}
\]

\[
u_{sw1} = (\tilde{\Delta} - \tilde{\chi} - \tilde{\rho}) \frac{s}{\|s\|}
\]

\[
\tilde{\Delta} > \varepsilon + \parallel \Delta \parallel, \tilde{\chi} > \varepsilon + \parallel \chi \parallel, \tilde{\rho} > \varepsilon + \parallel \rho \parallel, \varepsilon > 0
\]

The following Lyapunov function is introduced \( V_2 \)

\[
V_2 = V_1 + \frac{1}{2} s^T s
\]

We can obtain \( \dot{V}_2 \):

\[
\dot{V}_2 = -k_s s_i^2 + s^T [M(q)^{-1}(\Delta - \chi - \rho) - \parallel M(q)^{-1} \parallel (\tilde{\Delta} - \tilde{\chi} - \tilde{\rho}) \frac{s}{\parallel s\|}] \\ 
\leq -k_s s_i^2 + s^T \parallel s \parallel \parallel M^{-1}(q) \parallel \parallel \Delta - \chi - \rho \parallel - (\tilde{\Delta} - \tilde{\chi} - \tilde{\rho}) \\ 
\leq -k_s s_i^2 + 3 \parallel s \parallel \parallel M^{-1}(q) \parallel \varepsilon \\ 
\leq 0
\]

In order to reduce chattering, the switching term of the above formula (22) is replaced by the switching term with saturation function.

\[
u_{sw2} = (- - -) \text{sgn}(s)
\]

\[
\text{sgn}(s) = \frac{\exp(s) - \exp(-s)}{\exp(s) + \exp(-s)}
\]

3.3. Controller with Neural Network Adaptive

The RBF neural network can realize effective compensation under the condition of limited actuator output. It can also compensate for the upper bound of system uncertainty error. Thus, integrate the two terms into \( \tilde{\nu} \) and approach them through RBF neural network. From (2), it follows that:

\[
\tilde{\nu} = \tilde{W}^T H(X) + \chi(x)
\]

Adding neural network adaptive method, we can design a new controller
\[ u_2 = C(q, \dot{q}) \dot{q} + G(q) + \dot{p}(t)[c_1 \dot{s}_1(0) + \dot{s}_2(0)] - \tau_f + M(q) [\ddot{\alpha} - c_2 \dot{s}_1 - \frac{s}{\|s\|^2} (s_1 s_2)] + u_{eq} \]  

(28)

\[ u_{eq} = (\hat{W}^TH(x)) \text{sgn}(s) \]  

(29)

The network updating law is given by

\[ \hat{W} = \xi H(X)s(t)^TM^{-1}(q) \]  

(30)

Consider the Lyapunov function candidate

\[ V = V_2 + V_3 \]  

(31)

\[ V_3 = \frac{1}{2} \text{tr} [\hat{W}^T \xi^{-1} \hat{W}(t)], \xi > 0 \]  

(32)

Define weight estimation error \( \hat{W} = W^* - \hat{W} \), since [17], we can change the form of the derivative of \( V_3 \):

\[
\dot{V}_3 = \text{tr} [\hat{W}^T \xi^{-1} (-\xi H(X)s(t)^TM^{-1}(q))] \\
= -2\text{tr} [\hat{W}H(x)s(t)^TM^{-1}(q)] \\
= -s(t)^TM(q)^{-1}\hat{W}^TH(x) 
\]

Further, the time derivative of \( V \) is:

\[
\dot{V} = -k_1 s_1^2 + s^T [M(q)^{-1}(\Delta - \chi - \rho)] - s^T [M(q)^{-1}\hat{W}^TH(X)] + V_3 \\
= -k_1 s_1^2 + s^T [M(q)^{-1}(\Delta - \chi - \rho)] - s^T [M(q)^{-1}\hat{W}^TH(X)] - s^T M(q)^{-1}\hat{W}^TH(x) \\
= -k_1 s_1^2 + s^T [M(q)^{-1}(\hat{W}^TH(X) + \epsilon(x)) - \hat{W}^TH(X)] - s^T M(q)^{-1}\hat{W}^TH(x) \\
= -k_1 s_1^2 - s^T M(q)^{-1}(\hat{W}^TH(x) + \epsilon) - s^T M(q)^{-1}\hat{W}^TH(x) \\
= -k_1 s_1^2 - s^T M(q)^{-1} \epsilon \\
\leq 0
\]

Thus, it is proved that the method can guarantee the global asymptotic stability of the slow subsystem.

4. System Simulation

In this section, to illustrate the effectiveness of the proposed approach, choose model parameters of 2-joint FJ manipulator as

\[
M_m(q) = \begin{bmatrix}
0.1 + 0.01\cos q_2 & 0.01\sin q_2 \\
0.01\sin q_2 & 0.1
\end{bmatrix}
\]

\[
C(q, \dot{q}) = \begin{bmatrix}
-0.005\sin q_2 \dot{q}_2 & 0.005\cos q_2 \dot{q}_2 \\
0.005\cos q_2 \dot{q}_2 & 0.005\sin q_2 \dot{q}_2
\end{bmatrix}
\]

\[
G(q) = \begin{bmatrix}
0.01g\cos(q_1 + q_2) \\
0.01g\cos(q_1 + q_2)
\end{bmatrix}, g = 9.8
\]

\[
\chi + \rho = \begin{bmatrix}
0.2\cos(t) \\
0.2\sin(t)
\end{bmatrix}, J(q) = \begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix}, K = \begin{bmatrix}
100 & 0 \\
0 & 100
\end{bmatrix}
\]

In addition, the desired trajectories are given as \( r_1 = \sin(t) \) and \( r_2 = \cos(t) \). The initial states are:

\[
[q_1 \quad \dot{q}_1 \quad q_2 \quad \dot{q}_2] = [1 \quad 0 \quad 0 \quad 0]
\]

Select parameters: \( \vartheta = 2 \).
\[ \lambda = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix} , k_i = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix} , c_i = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix} , k_f = \begin{bmatrix} 10 & 0 \end{bmatrix} \]

To achieve good performance, selecting five neurons in hidden layer of RBF neural network, the parameters of RBF neural network are

\[ c_j = 0.6 \begin{bmatrix} -1 & -0.5 & 0 & 0.5 & 1 \\ -1 & -0.5 & 0 & 0.5 & 1 \end{bmatrix} , b_j = 0.5 [1 \ 1 \ 1 \ 1 \ 1] \]

The following Figures is the simulation results of the control of the manipulator with flexible joints by the method proposed in this paper. Figure 1 is the trajectory tracking diagram of two joints. At about 0.75 seconds, the reference track can be tracked under a big initial error, which reflects the rapidity and effectiveness of the controller. Figure 2 is the diagram of system speed tracking. In Figure 3 rad, the chattering of the controller is nearly eliminated, but there will be an explode signal in the initial state, which leads to excessive control torque. However, it is impossible to realize the controller completely in practice. Figure 4 shows the tracking error of the trajectory and its enlarged view. It can be seen that the error is controlled on the order of \(10^{-3}\) rad, which evinces the accuracy of the controller. Figure 5 shows the simulation results when the fast control law is closed. According to the results, we can see the effectiveness of the control law of the fast on suppression of flexibility. Figures 6 and 7 show the simulation results of trajectory tracking and control torque when the upper and lower bounds of the execution torque saturation given as 5 N. The control torque is restricted in a delimited range. The results show that even though the actuator is not able to realize the output signal of the controller, it can still track the trajectory, which again reflects the effectiveness that RBF neural network can compensation the actuator output saturation.

![Figure 1](image1.png)
(a) joint 1
(b) joint 2

**Figure 1. Trajectory tracking.**

![Figure 2](image2.png)
(a) joint 1
(b) joint 2

**Figure 2. Speed tracking.**

Figure 3. Actuator output.

Figure 4. Tracking error.

Figure 5. Close fast subsystem controller trajectory tracking.
Figure 6. Execution saturation 5 trajectory tracking.

(a) joint 1

(b) joint 2

Figure 7. Execution saturation 5 actuator output.

Figure 8 and Figure 9 are the simulation results of the traditional sliding mode based on the approach of singular perturbation. The fast controller is also chosen as the velocity-difference-based feedback method, and the sliding mode surface is the linear one:

\[ s = c_1 \dot{e} + \dot{\dot{e}} \] (33)

The slow control law is:

\[ u_s = C(q, \dot{q}) \dot{\dot{q}} + G(q) + M[\ddot{q}_d - c_1 \dot{e} - \eta \text{sign}(s)] \] (34)

Choosing \( \eta = 5 \), \( \eta \) should suffice to ensure the stability of the system, but as \( \eta \) increases, the chattering of the controller will soar, although the velocity of the system increases as well. Under the same parameters, it can be seen clearly from the simulation results that the traditional linear sliding mode controller follows the trajectory in about 1.5 seconds, and strong chattering occurred after the system reaches the sliding mode surface, which is impractical in reality.
5. Conclusion
Based on the singular perturbation method, this paper proposes an approach of backstepping global sliding mode. To verify the method, many simulations are completed to address the problem of trajectory tracking of FJ manipulator, which is brought by the inaccuracy of the model and the limited actuator output torque. The results suggest that joint flexibility can be restrained by the control law of the fast subsystem. On the other hand, the backstepping global sliding mode controller with nonlinear integral surfaces can track the reference trajectory rapidly and accurately, it's remarkable that the signal of the controller is nearly chattering-free. With the output saturation, the neural network adaptive law designed in this paper is effective in compensating the unexecuted torque. In addition, detailed and rigorous proof of its stability is given. The theoretical analysis and simulation results all suggest that the controller designed in this paper is effective.

References
[1] Guo Q, Li X, and Jiang D. 2018 Full-state error constraints based dynamic surface control of electro-hydraulic system IEEE Access 6 53092-101
[2] Mondal S and Mahanta C 2014 Adaptive second order terminal sliding mode controller for robotic manipulators Journal of the Franklin Institute 351(04) 2356-77
[3] Yang Y, Dong X, Liu X, and Huang D 2020 Robust repetitive learning-based trajectory tracking control for a Leg Exoskeleton driven by hybrid hydraulic System IEEE Access 8 27705-14
[4] Rsetam K, Cao Z, and Man Z 2016 Hierarchical sliding mode control applied to a single-link flexible joint robot manipulator 2016 Intl Conf on Advanced Mechatronic Systems (Melbourne-VIC) pp 476-81
[5] Rsetam K, Cao Z, Man Z, and Mitrevska M 2017 Optimal second order integral sliding mode control for a flexible joint robot manipulator IECON 2017 - 43rd Annual Conf of the IEEE Industrial Electronics Society(Beijing) pp 3069-74
[6] Chen J, and Li R 2013 Sliding mode observer design for flexible joint manipulators Fifth Intl Conf on Intelligent Human-Machine Systems and Cybernetics 2013 338-342
[7] Yang J, Shao R, Li B, et al 2018 Iterative learning observer based robust output feedback tracking control for flexible manipulator Harbin Gongcheng Daxue Xuebao/Journal of Harbin Engineering University 39 106-13
[8] Dian S, Hu Y, Zhao T, et al 2019 Adaptive backstepping control for flexible-joint manipulator using interval type-2 fuzzy neural network approximator, Nonlinear Dynamics 97(02)
[9] Hong-Rui W, Yu-Dong F, and Xiu-Ling L 2010 Adaptive dynamic sliding mode control for robotic manipulator based on backstepping Computer Engineering and Applications 46(08) 211-213
[10] Zhi-Guo T, Zhe L, Xin-Bo W, et al 2017 RBF neural network sliding mode control of onboard craning manipulator based on backstepping 2017 Chinese Automation Congress (CAC) IEEE(Jinan) pp 2226-2231

[11] Yang Y, Huang D, and Dong X 2019 Enhanced neural network control of lower limb rehabilitation exoskeleton by add-on repetitive learning Neurocomputing 323 256-264

[12] Khan Q, and Akmeliawati R 2017 Neuro-adaptive dynamic integral sliding mode control design with output differentiation observer for uncertain higher order MIMO nonlinear systems, Neurocomputing 226

[13] Hong-Feng T, Xue-Hui X, and Hui-Zhong Y 2014 Periodic adaptive compensating learning control of nonlinear systems with saturated input Acta Automatica Sinica 40 9 1998-2004

[14] Ma J, Jin D, Wei Z, et al 2018 Boundary control of a flexible manipulator based on a high order disturbance observer with input saturation Shock and vibration 208 64241-410

[15] He W, Dong Y 2016 Adaptive neural impedance control of a robotic manipulator with input saturation IEEE Transactions on Systems, Man, and Cybernetics: Systems 46(3) 334-44

[16] Spong M W 1995 Adaptive control of flexible joint manipulators: comments on two papers Automatica 31(4) 585-90

[17] Feng G 1995 A compensating scheme for robot tracking based on neural networks Robot Auton Syst 15 199-206