

Chattering-Free Neuro-Sliding Mode Control of 2-DOF Planar Parallel Manipulators

Regular Paper

Tien Dung Le¹, Hee-Jun Kang^{2,*} and Young-Soo Suh²

¹ Graduate School of Electrical Engineering, University of Ulsan, Ulsan, South Korea

² School of Electrical Engineering, University of Ulsan, Ulsan, South Korea

* Corresponding author E-mail: hjkang@ulsan.ac.kr

Received 20 Aug 2012; Accepted 16 Nov 2012

DOI: 10.5772/55102

© 2013 Le et al.; licensee InTech. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract This paper proposes a novel chattering free neuro-sliding mode controller for the trajectory tracking control of two degrees of freedom (DOF) parallel manipulators which have a complicated dynamic model, including modelling uncertainties, frictional uncertainties and external disturbances. A feedforward neural network (NN) is combined with an error estimator to completely compensate the large nonlinear uncertainties and external disturbances of the parallel manipulators. The online weight tuning algorithms of the NN and the structure of the error estimator are derived with the strict theoretical stability proof of the Lyapunov theorem. The upper bound of uncertainties and the upper bound of the approximation errors are not required to be known in advance in order to guarantee the stability of the closed-loop system. The example simulation results show the effectiveness of the proposed control strategy for the tracking control of a 2-DOF parallel manipulator. It results in its being chattering-free, very small tracking errors and its robustness against uncertainties and external disturbances.

Keywords Parallel Manipulators, Neural Network, Sliding Mode Control, Chattering-Free, Uncertainty Compensator

1. Introduction

Parallel manipulators are closed-loop kinematic chain mechanisms which have such advantages as high accuracy, high stiffness, high payload capability and low moving inertia, etc. They are widely used in numerous applications, such as humanoid robots, simulators, medical robots and micro-robots, and they are becoming increasingly popular in the machine-tool industry [1]. Compared to serial manipulators, the dynamic model of parallel manipulators is significantly complicated by the presence of multiple closed-loop chains and singularities. As a result, the control of parallel manipulators needs more advanced control techniques than that of serial manipulators.

The motion control of parallel manipulators has attracted many researchers in studying its potential performance.

A proportional derivative (PD) controller [2], a nonlinear PD controller [3] and an adaptive switching learning PD control method (ASL-PD) [4] were proposed for the motion control of parallel manipulators. All of these controllers are simple and easy to implement but they do not perform well because the full dynamics of the robots are not considered and compensated for. In [5-7], the synchronization controllers were presented for parallel manipulators. This kind of controller - also only based on the parallel manipulator's kinematics - can improve the performances of the trajectory tracking further, but it requires more complicated computation. Some other advanced controllers were proposed, such as the computed torque controller [8-11], the model-based iterative learning controller [12] and the adaptive controller [13]. These approaches are based on having full knowledge of the dynamic model of the robot and require heavy computational power. However, it is impossible to obtain a precise dynamic model of the parallel manipulators, due to the presence of multiple closed-loop chains, singularities, structured and unstructured uncertainties and external disturbances. Hence, there is a need for control strategies for parallel manipulators with robustness, adaptive capability, fast convergence and a simple structure.

Sliding mode control (SMC) has received much attention in the last few decades as a useful and powerful, robust control method in overcoming uncertainties, bounded external disturbances and unpredictable parameter variations [14, 15]. The theory of SMC has been successfully applied to serial manipulators [14, 16, 17] and parallel manipulators [18-21]. The main characteristic of SMC is the inclusiveness of a discontinuous control input which drives the control system towards a sliding surface $S(x,t)=0$ whereby the sliding mode happens along this surface. For handling large uncertainties (such as in the case of parallel manipulators) and external disturbances, a large gain of the discontinuous control input must be applied. For choosing the value of this gain, the upper bound of the uncertainties has to be known in advance. It is, however, not easy to estimate this bound of the uncertainties. In addition, a large switching gain is undesirable for the increased chance of input chattering, which leads to the high wear of the mechanism and the excitation of un-modelled high-frequency dynamics. The most common approach for chattering reduction is to define a boundary layer around the sliding surface and then use a continuous approximation of the switching control input within the boundary layer [14]. This approach can reduce the chattering effectively, but there is a tradeoff between asymptotic tracking and the elimination of chattering for the width of the boundary layer. A thicker boundary layer would reduce the chattering but make the tracking error bigger, and vice versa. Therefore, when applying the SMC for parallel manipulators, the issue of how to

eliminate chattering while achieving small tracking errors becomes an important topic.

In recent years, intelligent methods such as neural networks and fuzzy logic systems have been successfully applied in order to universally approximate the unknown dynamics and uncertainties of robotic manipulators. Many important adaptive NN-based and fuzzy logic-based SMC schemes have been proposed in which the discontinuous control of the conventional SMC is replaced by a NN or fuzzy compensators. These controllers consist of an equivalent continuous feedback control component and a component derived from the intelligent compensators for the compensation of uncertainties and external disturbances. If the uncertainties and external disturbances are sufficiently compensated, there is no need to use the discontinuous feedback control law to achieve the sliding mode and, therefore, the chattering phenomenon can be eliminated. For example, in [16] and [22], two similar adaptive fuzzy sliding mode controllers for serial robotic manipulators were proposed in which an adaptive single-input single-output (SISO) fuzzy logic system (FLS) or SISO radial basic function networks (RBFNs) were used to approximate the discontinuous part of the control signal. These controllers can eliminate the chattering phenomenon and they do not need knowledge of the upper boundary of the uncertainties. However, the typical SISO, FLS or SISO RBFN cannot approximate precisely the complicated, highly nonlinear uncertainties and external disturbances of robotic manipulators. Therefore, the bounds of the approximation errors have to be known for the chosen control gains of an auxiliary controller for enhancing the stability of the control system. In another paper, two kinds of adaptive SMC schemes for a serial robotic manipulator using a fuzzy compensator were proposed [23]. In these SMC schemes, the decomposition of the uncertainties' function is introduced and the properties of the uncertainties and the dynamics of the serial manipulators are considered. However, these schemes are still complicated and the number of fuzzy rules of each FLS is big. Moreover, in [24], an adaptive fuzzy SMC for affine nonlinear systems was developed and successfully applied to control a four-bar linkage system. A FLS is combined with a switching control to compensate the large uncertainties of the control system. However, this control method can only be applied to SISO dynamic systems and it seems that the chattering phenomenon still exists in their experimental results. Furthermore, an adaptive control of robot manipulators with the NN-based compensation of frictional uncertainties was proposed in [25]. The friction dynamic which is usually the cause of performance degradation at low velocities in the motion control of robotic manipulators is successfully modelled and compensated for. However, the control scheme did not consider the compensation of any modelling errors and

external disturbances of the robotic control system. On the other hand, a neuro-sliding mode control of robotic manipulators was presented in [26]. The controller used two parallel NNs - one for learning the continuous equivalent control and the other for the discontinuous control computation. Although the experimental studies of the paper showed good results, there is a lack of mathematical proof of stability. In addition, it is difficult to implement this controller with robotic manipulators, which have a complicated structure due to the large size of NNs and the enormous number of calculations.

Unlike serial manipulators, the dynamic behaviour of parallel manipulators is strongly nonlinear due to the highly dynamic coupling between the links. In addition, the number of links in parallel manipulators is often double or triple the number of links in serial manipulators with the same number of degrees of freedom. Therefore, the modelling errors and uncertainties in parallel manipulators are significantly larger and more complicated than in the case of serial manipulators. This makes it difficult to apply the above mentioned method which as proposed for serial manipulators to parallel ones. In this paper, we propose a novel SMC method for a tracking controller of 2-DOF parallel manipulators. First, the dynamic model of the 2-DOF parallel manipulators is analysed to build a continuous equivalent control. Next, a feedforward NN is combined with an error estimator to sufficiently compensate the modelling errors, uncertainties and external disturbances of the parallel manipulator system. The weights of the NN are adapted online and the proposed controller can guarantee the stability of the closed-loop system, overcome the chattering problem and improve the robustness of the control system. Compared with the existing controllers, the advantages of the control strategy proposed in this paper are twofold:

1. By using a NN combined with an error estimator, the large model uncertainties and external disturbances in the parallel manipulators' control system can be sufficiently compensated for. This is different from the existing methods using only a NN or a FLS for compensating the perturbation and ignoring the approximation errors or higher-order terms of Taylor's series expansion. This feature means that the proposed control method is suitable for application to the tracking control of parallel manipulators which have a complicated dynamic model and huge uncertainties.
2. The control strategy does not need to know the upper bound of any uncertainties and also does not need to know the bound of any approximation errors. The condition of the chosen control gains in the proposed controller is simple (it just needs a positive parameter and it does not need to know any threshold) but can guarantee the stability of the

closed-loop system. The structure of the NN and the error estimator, as well as the online learning algorithm of the NN in the proposed controller of this paper, are simple and easy to implement but lead to the acquisition of good results.

The rest of the paper is organized as follows. In section 2, the dynamic model of the 2-DOF planar parallel manipulators is formulated in the active joint space based on Lagrange-D'Alembert and the virtual work approach. In section 3, the architecture of the feedforward NN which is used in the proposed controller is presented. The proposed neuro-sliding mode controller for the trajectory tracking of 2-DOF planar parallel manipulators is presented in section 4. In section 5, simulations of trajectory tracking are carried out in order to verify the effectiveness of the proposed controller. Finally, several important remarks are concluded in section 6.

2. Dynamic model of 2-DOF planar parallel manipulators

In this section, we develop a dynamic model for a class of 2-DOF planar parallel manipulators acting on a horizontal plane and a reference frame is established in the workspace, as depicted in Figure 1. This kind of parallel manipulator consists of two active joints and three passive joints. The active joints are actuated by actuators while the passive joints are free to move.

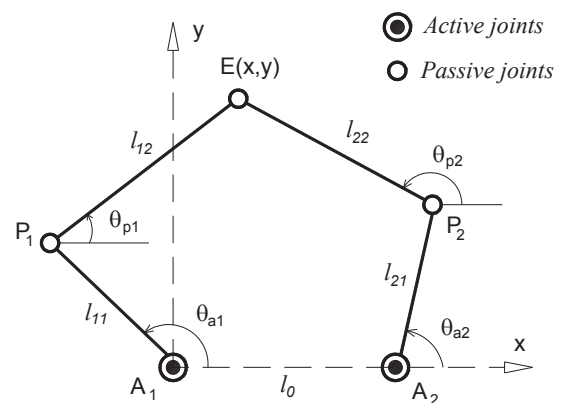


Figure 1. The 2-DOF planar parallel manipulator.

By $\theta_a = (\theta_{a1}, \theta_{a2})^T$ and $\theta_p = (\theta_{p1}, \theta_{p2})^T$ we denote the corresponding active joint vector and passive joint vector, respectively; $\mathbf{X} = (x, y)^T$ as the Cartesian coordinates; and $E(x, y)$ as the end-effector of the parallel manipulators. The link lengths of the parallel manipulators are $l_{11} = A_1P_1$, $l_{12} = P_1E$, $l_{21} = A_2P_2$, $l_{22} = P_2E$ and $l_0 = A_1A_2$.

The 2-DOF planar parallel manipulators have one closed-loop. Suppose that this closed-loop is virtually cut at the common point E and forms an equivalent tree-structure open kinematic chain, including two independent 2-DOF serial manipulators, as shown in Figure 2. It is assumed that in this open kinematic chain all of the passive joints are actuated by virtual actuators. Suppose that the

equivalent tree-structure makes the same motion as the original closed-chain without force or moment interaction at the virtually cut joint.

The dynamic model of each planar 2-DOF serial manipulator in the tree-structure is given by Lagrange's equation [27]:

$$\frac{d}{dt} \left(\frac{\partial L_i}{\partial \dot{\theta}_i} \right) - \frac{\partial L_i}{\partial \theta_i} = \tau_i - F_i, \quad i = 1, 2 \quad (1)$$

where L_i is the Lagrangian function; $\theta = (\theta_{ai}, \theta_{pi})^T$ is the joint vector; $\tau = (\tau_{ai}, \tau_{pi})^T$ is the joint torque vector; and $F_i = (f_{ai}, f_{pi})^T$ is the vector of the active and passive joint friction torques of the i^{th} serial chain.

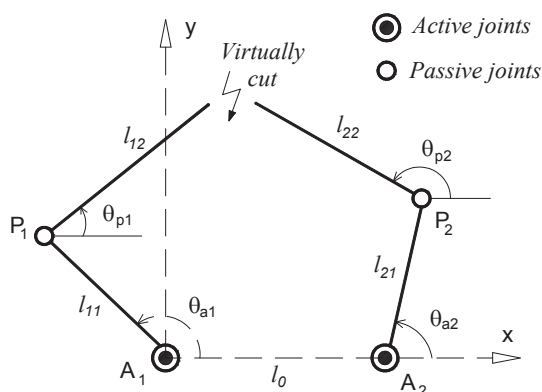


Figure 2. Tree-structure system of the cutting joint.

As the parallel manipulator operates on a horizontal planar plane, the Lagrangian functions only contain the kinetic energy of the mechanism:

$$L_i = \frac{1}{2} m_{i1} (\dot{x}_{i1}^2 + \dot{y}_{i1}^2) + \frac{1}{2} I_{zi1} \dot{\theta}_{ai}^2 + \frac{1}{2} m_{i2} (\dot{x}_{i2}^2 + \dot{y}_{i2}^2) + \frac{1}{2} I_{zi2} \dot{\theta}_{pi}^2 \quad (2)$$

where m_{i1} and m_{i2} are the masses of the links of the serial chain i ; I_{zi1} , I_{zi2} are the inertia tensors of the links of the serial chain i ($i=1,2$).

Letting r_{i1} and r_{i2} be the distance from the joints to the centre of mass for each link, we have:

$$\begin{aligned} \dot{x}_{i1} &= -r_{i1} \sin \theta_{ai} \dot{\theta}_{ai}, \\ \dot{y}_{i1} &= r_{i1} \cos \theta_{ai} \dot{\theta}_{ai}, \\ \dot{x}_{i2} &= -l_{i1} \sin \theta_{ai} \dot{\theta}_{ai} - r_{i2} \sin \theta_{pi} \dot{\theta}_{pi}, \\ \dot{y}_{i2} &= l_{i1} \cos \theta_{ai} \dot{\theta}_{ai} + r_{i2} \cos \theta_{pi} \dot{\theta}_{pi}. \end{aligned}$$

By substituting the above equations into (2), the Lagrangian function becomes:

$$L_i = \frac{1}{2} \left[\alpha_i \dot{\theta}_{ai}^2 + 2\beta_i \cos(\theta_{ai} - \theta_{pi}) \dot{\theta}_{ai} \dot{\theta}_{pi} + \delta_i \dot{\theta}_{pi}^2 \right] \quad (3)$$

in which $\alpha_i = m_{i1}r_{i1}^2 + I_{zi1} + m_{i2}l_{i1}^2$, $\beta_i = m_{i2}l_{i1}r_{i2}$, and $\delta_i = m_{i2}r_{i2}^2 + I_{zi2}$ correspond to the dynamic parameters, $i = 1, 2$.

Substituting the Lagrange function (3) into Lagrange's equation (1), we have the dynamic equations of each serial chain of a tree-structure as:

$$M_i(\theta_i) \ddot{\theta}_i + C_i(\theta_i, \dot{\theta}_i) \dot{\theta}_i + F_i = \tau_i, \quad i = 1, 2 \quad (4)$$

where $M_i \in \mathbb{R}^{2 \times 2}$ is the inertia matrix and $C_i \in \mathbb{R}^{2 \times 2}$ is the Coriolis and centrifugal force matrix, which are defined as:

$$M_i(\theta_i) = \begin{bmatrix} \alpha_i & \beta_i c_{api} \\ \beta_i c_{api} & \delta_i \end{bmatrix}, \quad (5)$$

$$C_i(\theta_i, \dot{\theta}_i) = \begin{bmatrix} 0 & \beta_i s_{api} \dot{\theta}_{pi} \\ -\beta_i s_{api} \dot{\theta}_{ai} & 0 \end{bmatrix} \quad (6)$$

where the symbols c_{api} and s_{api} are, respectively, defined as: $c_{api} = \cos(\theta_{ai} - \theta_{pi})$, $s_{api} = \sin(\theta_{ai} - \theta_{pi})$, $i = 1, 2$.

Combining the dynamic models of two open serial chains together, the dynamic model of the equivalent tree-structure mechanism is obtained as:

$$M_t \ddot{\theta} + C_t \dot{\theta} + F_t = \tau_t \quad (7)$$

in which $\theta = (\theta_a, \theta_p)^T \in \mathbb{R}^4$ is the vector of the joint angles; $\tau_t = (\tau_a, \tau_p)^T \in \mathbb{R}^4$ is the joint vector; and $F_t = (F_a, F_p)^T \in \mathbb{R}^4$ is the friction torque vector of the equivalent tree-structure open chain system. $\tau_a = (\tau_{a1}, \tau_{a2})^T$ and $\tau_p = (\tau_{p1}, \tau_{p2})^T = (0, 0)^T$ are the input torque vectors of the active joints and passive joints respectively. $F_a = (f_{a1}, f_{a2})^T$ and $F_p = (f_{p1}, f_{p2})^T = (0, 0)^T$ are the friction force vectors of the active joints and passive joints respectively. Here, we assume that the effect of the friction force on the passive joints is much smaller than that on the active joints. Thus, in order to simplify the dynamic model, only the disturbances on the active joints are considered.

The inertia matrix $M_t \in \mathbb{R}^{4 \times 4}$ and the Coriolis and centrifugal force matrix $C_t \in \mathbb{R}^{4 \times 4}$ in (7) are expressed by:

$$M_t = \begin{bmatrix} \alpha_1 & 0 & \beta_1 c_{ap1} & 0 \\ 0 & \alpha_2 & 0 & \beta_2 c_{ap2} \\ \beta_1 c_{ap1} & 0 & \delta_1 & 0 \\ 0 & \beta_2 c_{ap2} & 0 & \delta_2 \end{bmatrix}, \quad (8)$$

$$C_t = \begin{bmatrix} 0 & 0 & \beta_1 s_{ap1} \dot{\theta}_{p1} & 0 \\ 0 & 0 & 0 & \beta_2 s_{ap2} \dot{\theta}_{p2} \\ -\beta_1 s_{ap1} \dot{\theta}_{a1} & 0 & 0 & 0 \\ 0 & -\beta_2 s_{ap2} \dot{\theta}_{a2} & 0 & 0 \end{bmatrix} \quad (9)$$

Next, the loop constraints are taken into account using a Jacobian matrix. From D'Alembert's principle and the principle of virtual work, the active joint torques τ_a and the generalized torques τ satisfy the following equation [28]:

$$\tau_a = \Psi^T \tau_t \quad (10)$$

where $\Psi = \partial \theta / \partial \theta_a \in \mathbb{R}^{4 \times 2}$ is the Jacobian matrix of all the joints with respect to the active joints. We have: $\Psi = [\mathbf{I}, \mathbf{J}]^T$ where $\mathbf{I} \in \mathbb{R}^{2 \times 2}$ is identity matrix and $\mathbf{J} = \partial \theta_p / \partial \theta_a \in \mathbb{R}^{2 \times 2}$ is computed from the loop constraint equations:

$$\mathbf{J} = \frac{\partial \theta_p}{\partial \theta_a} = - \left[\frac{\partial \mathbf{h}}{\partial \theta_p} \right]^{-1} \left[\frac{\partial \mathbf{h}}{\partial \theta_a} \right] \quad (11)$$

$$\mathbf{h} = \begin{bmatrix} h_x \\ h_y \end{bmatrix} = \begin{bmatrix} l_{11}c_{a1} + l_{12}c_{p1} - l_0 - l_{21}c_{a2} - l_{22}c_{p2} \\ l_{11}s_{a1} + l_{12}s_{p1} - l_{21}s_{a2} - l_{22}s_{p2} \end{bmatrix} \quad (12)$$

where the symbols c_{ai} , s_{ai} and c_{pi} , s_{pi} are, respectively, defined as: $c_{ai} = \cos \theta_{ai}$, $s_{ai} = \sin \theta_{ai}$, $c_{pi} = \cos \theta_{pi}$, $s_{pi} = \sin \theta_{pi}$, $i = 1, 2$.

Taking the constraint (10) into the tree-structure by multiplying both sides of equation (7) by Ψ^T , we obtain:

$$\Psi^T \mathbf{M}_t \ddot{\theta} + \Psi^T \mathbf{C}_t \dot{\theta} + \Psi^T \mathbf{F}_t = \Psi^T \tau_t \quad (13)$$

In addition, we have the following relationships:

$$\dot{\theta} = \frac{\partial \theta}{\partial \theta_a} \dot{\theta}_a \quad (14)$$

$$\ddot{\theta} = \dot{\Psi} \dot{\theta}_a + \Psi \ddot{\theta}_a \quad (15)$$

By substituting (10), (14) and (15) into (13), we obtain the dynamic model of the 2-DOF planar parallel manipulators in the active joints space:

$$\hat{\mathbf{M}}_a \ddot{\theta}_a + \hat{\mathbf{C}}_a \dot{\theta}_a + \mathbf{F}_a = \tau_a \quad (16)$$

where $\hat{\mathbf{M}}_a = \Psi^T \mathbf{M}_t \Psi \in \mathbb{R}^{2 \times 2}$ is the inertia matrix and $\hat{\mathbf{C}}_a = \Psi^T \mathbf{M}_t \dot{\Psi} + \Psi^T \mathbf{C}_t \Psi \in \mathbb{R}^{2 \times 2}$ is the Coriolis and centrifugal matrix.

The dynamic model (16) has the following properties, which are proved in [29]:

Property 1: $\hat{\mathbf{M}}_a$ is symmetric and positive definite.

Property 2: $\dot{\hat{\mathbf{M}}}_a - 2\hat{\mathbf{C}}_a$ is skew-symmetric.

Because of the presence of the highly nonlinear uncertainties, the exact dynamic model of the parallel

robot will never be known. If the modelling errors caused by the uncertainties are bounded, we can express the actual dynamics by combining the estimated dynamics with the modelling errors as the following equation:

$$\mathbf{M}_a \ddot{\theta}_a + \mathbf{C}_a \dot{\theta}_a + \mathbf{F}_a = \tau_a \quad (17)$$

where $\mathbf{M}_a = \hat{\mathbf{M}}_a + \Delta \mathbf{M}_a$ and $\mathbf{C}_a = \hat{\mathbf{C}}_a + \Delta \mathbf{C}_a$ are the actual dynamic parameters of the parallel manipulators; and $\Delta \mathbf{M}_a$ and $\Delta \mathbf{C}_a$ are the bounded modelling errors.

Consider the external disturbances which affect the active joints; we define the vector of unknown uncertainties and external disturbances as:

$$\Delta \tau_a = \Delta \mathbf{M}_a \ddot{\theta}_a + \Delta \mathbf{C}_a \dot{\theta}_a + \mathbf{F}_a + \mathbf{d}(t) \quad (18)$$

where $\mathbf{d}(t) \in \mathbb{R}^2$ is the vector of the external disturbances.

From (17) and (18), we obtain the actual dynamic equation of the 2-DOF planar parallel manipulators in the active joint space:

$$\hat{\mathbf{M}}_a \ddot{\theta}_a + \hat{\mathbf{C}}_a \dot{\theta}_a + \Delta \tau_a = \tau_a \quad (19)$$

The dynamic equation (19) is very useful for the analysis, simulation study and control design of the 2-DOF planar parallel manipulators.

Although the dynamic model (19) of the 2-DOF parallel manipulators in the active joint space has a similar form to the model (4) of the serial manipulators, its dynamic behaviour is much more complicated and strongly nonlinear due to the highly dynamic coupling between the links. We can see that the nominal parameter matrices in equation (16) are much more complicated in comparison with the dynamic model of serial manipulators. In addition, the uncertainties $\Delta \tau_a$ in the dynamic model (18) are huge and highly nonlinear due to the large number of links, the loop constraints, the friction and the variation of the parameters. Thus, it is difficult to reuse the existing control strategies which were proposed for serial manipulators in the literature for improving the performance of the tracking control of the parallel manipulators.

3. The feedforward neural network architecture

The NN used in this paper has the structure indicated in Figure 3. The architecture of the NN includes the input layer, the hidden layer and the output layer. The NN has 2 outputs corresponding to the 2 active joints of the parallel manipulators considered.

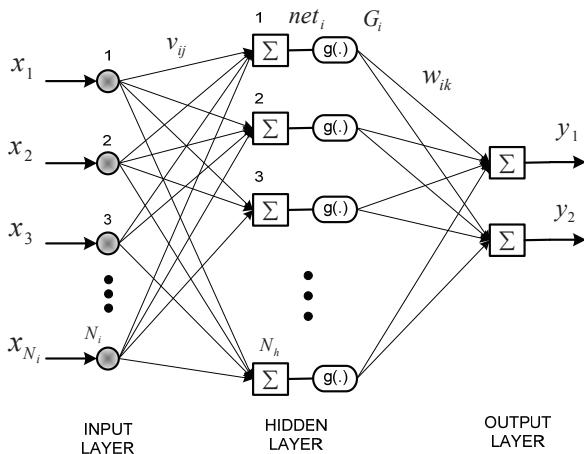


Figure 3. Structure of the feedforward neural network.

The input layer: The input vector of the NN is denoted by:

$$x = [x_1, x_2, \dots, x_{N_i}]^T \quad (20)$$

where N_i represents the number of components of the input vector.

The hidden layer: By denoting the number of neurons in the hidden layer as N_h , the weight matrix connecting the input and the hidden layers is expressed by:

$$V = (v_1, v_2, \dots, v_{N_h}) \in \mathbb{R}^{N_i \times N_h},$$

$$v_i = (v_{i1}, v_{i2}, \dots, v_{iN_i})^T \in \mathbb{R}^{N_i}, i = \overline{1, N_h} \quad (21)$$

The inputs and outputs of the hidden layer are, respectively, presented as:

$$net_i = \sum_{j=1}^{N_i} v_{ij} x_j, \quad (22)$$

$$G_i = g(net_i), \quad i = \overline{1, N_h}. \quad (23)$$

The transfer function in the hidden layer is used as a sigmoid function:

$$g(z) = \frac{1 - \exp(-z)}{1 + \exp(-z)} \quad (24)$$

The output layer: The weight matrix connecting the hidden and output layers is expressed by:

$$W = (w_1, w_2, \dots, w_{N_h})^T \in \mathbb{R}^{N_h \times 2},$$

$$w_i = (w_{i1}, w_{i2}) \in \mathbb{R}^2, i = \overline{1, N_h} \quad (25)$$

The outputs of the NN are expressed by:

$$y_k = \sum_{i=1}^{N_h} w_{ik} G_i, \quad k = 1, 2 \quad (26)$$

The outputs of the NN can be represented in a vector form:

$$y = W^T G(x, V) \in \mathbb{R}^2 \quad (27)$$

where $G = [G_1, G_2, \dots, G_{N_h}]^T \in \mathbb{R}^{N_h \times 1}$.

4. The proposed controller

4.1 Traditional Sliding Mode Controller

Let $\theta_{da} = (\theta_{da1}, \theta_{da2})^T$ be the desired state vector and $e = \theta_a - \theta_{da}$ be the tracking error vector of the parallel manipulators. The first step in the design of the sliding mode control for the system (19) is to design the sliding surface function as:

$$s = \dot{e} + \Lambda e = \dot{\theta}_a - (\theta_{da} - \Lambda e) = \dot{\theta}_a - \dot{\theta}_{ar} \quad (28)$$

where $\Lambda = \text{diag}(\lambda_1, \lambda_2)$, λ_1 and λ_2 are positive constants which determine the motion feature in the sliding surface; and $\dot{\theta}_{ar} = \dot{\theta}_{da} - \Lambda e$ is defined as the reference velocity vector.

In the second step, a control law $\tau_a \in \mathbb{R}^2$ is designed such that the system state trajectories are driven to the sliding surface and kept on the sliding surface. The reaching condition is expressed by [14]:

$$\frac{1}{2} \frac{d}{dt} s_i^2 \leq -\zeta_i |s_i|, \quad i=1, 2 \quad (29)$$

where ζ_i is a strictly positive constant. Equation (29) indicates that the energy of s should decay so long as s is not zero.

In general, the control input τ_a consists of two components:

$$\tau_a = \tau_{eq} + \tau_{sw} \quad (30)$$

where the first term $\tau_{eq} \in \mathbb{R}^2$ is the *equivalent control* which keeps the trajectory of the system state on the sliding surface; and the second term $\tau_{sw} \in \mathbb{R}^2$ is the discontinuous control which drives the system states toward the sliding surface when they are deviating from this surface.

The equivalent control is considered for the nominal system in the absence of the uncertainties and external disturbances:

$$\tau_{eq} = \hat{M}_a \ddot{\theta}_{ar} + \hat{C}_a \dot{\theta}_{ar} \quad (31)$$

The discontinuous control is designed as:

$$\tau_{sw} = -K \text{sign}(s) \quad (32)$$

where $K = \text{diag}(k_1, k_2)$, k_1 and k_2 are positive constants.

Now, by substituting (31) and (32) into (30), the conventional sliding mode controller for the parallel manipulators' dynamic model (19) is presented by:

$$\tau_a = \hat{M}_a \ddot{\theta}_{ar} + \hat{C}_a \dot{\theta}_{ar} - K \text{sign}(s) \quad (33)$$

It has been proven in [19] that by considering the Lyapunov function candidate as:

$$V = \frac{1}{2} s^T \hat{M}_a s \quad (34)$$

and choosing the switching gain matrix K such that $k_i \geq |\Delta \tau_{ai}|_{\text{bound}}$, where $|\Delta \tau_{ai}|_{\text{bound}}$ is the upper boundary of $|\Delta \tau_{ai}|$, the overall system is asymptotically stable. Thus,

the decay of the energy of s - as long as $s \neq 0$ - is guaranteed and the reaching condition (29) is satisfied.

4.2 Chattering Free Neuro-Sliding Mode Controller

The sign function in the discontinuous control term of the SMC (33) leads to the chattering problem. Therefore, in this section we propose a new controller in which the discontinuous control is replaced by a NN combined with an error estimator. The connection weights of the NN are adapted online and the structure of the error estimator f_{est} is designed with the strict theoretical stability proof of the Lyapunov theorem. The structure of the overall system is presented in Figure 4.

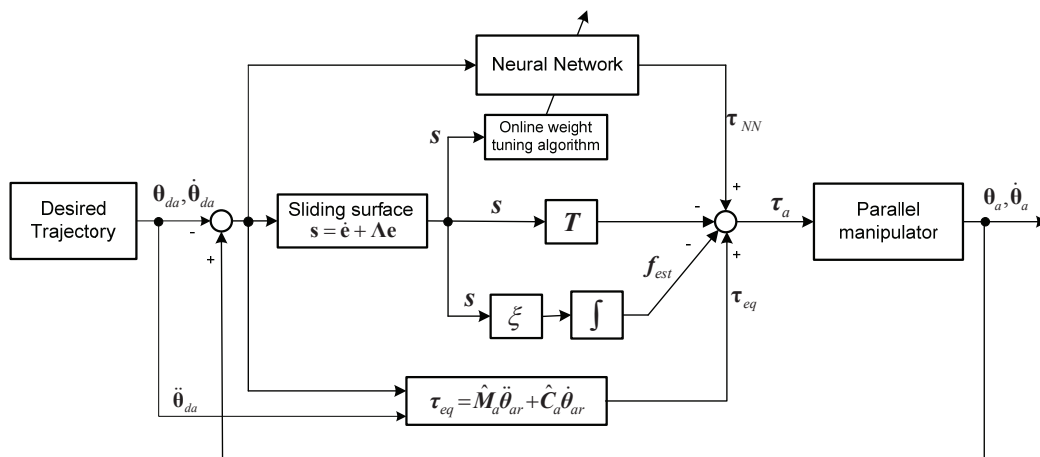


Figure 4. Block diagram of the proposed controller.

The proposed controller is expressed by the following equation:

$$\tau_a = \hat{M}_a \ddot{\theta}_{ar} + \hat{C}_a \dot{\theta}_{ar} + f_{NN} + f_{\text{est}} - Ts \quad (35)$$

where $f_{NN} \in \mathbb{R}^2$ is the output of the NN whose structure is described in section 3 for the online approximation of the unknown vector $\Delta \tau_a \in \mathbb{R}^2$. Since the output of the NN is not able to approximate $\Delta \tau_a$ accurately, the error estimator $f_{\text{est}} \in \mathbb{R}^2$ in (56) is used to attenuate the approximation errors. The term Ts is used in enhancing the robustness of the control system. Moreover, $T = \text{diag}(T_1, T_2)$ is a diagonal positive definite matrix in which T_1 and T_2 are positive constants.

The inputs of the NN are chosen as the errors and derivative of errors: $x = [e_1, \dot{e}_1, e_2, \dot{e}_2]^T \in \mathbb{R}^4$. The NN is used to approximate the unknown uncertainties $\Delta \tau_a$ online. The approximation error is denoted as:

$$\Delta \tau_a - f_{NN} = W^{*T} G(x, V^*) - \hat{W}^T G(x, \hat{V}) + \varepsilon_N \quad (36)$$

where $W^* \in \mathbb{R}^{N_h \times 2}$ and $V^* \in \mathbb{R}^{4 \times N_h}$ are the optimal values of the weight matrices W and V ; $\hat{W} \in \mathbb{R}^{N_h \times 2}$ and $\hat{V} \in \mathbb{R}^{4 \times N_h}$ are the estimates of the optimal weight matrices (W^* and

V^*); $\varepsilon_N \in \mathbb{R}^2$ is the bounded reconstruction error due to the inadequate number of neurons in the hidden layer of the NN.

For convenience, equation (36) is rewritten as:

$$\Delta \tau_a - f_{NN} = W^{*T} \tilde{G} + \tilde{W}^T \hat{G} + \varepsilon_N \quad (37)$$

where the symbols G^* , \hat{G} , \tilde{G} and \tilde{W} are, respectively, defined as $G^* \equiv G(x, V^*) \in \mathbb{R}^{N_h}$, $\hat{G} \equiv G(x, \hat{V}) \in \mathbb{R}^{N_h}$, $\tilde{G} = G^* - \hat{G} \in \mathbb{R}^{N_h}$ and $\tilde{W} = W^* - \hat{W} \in \mathbb{R}^{N_h \times 2}$.

The Taylor series expansion of \tilde{G} for a given x can be written as follows:

$$\begin{aligned} \tilde{G} &= \begin{bmatrix} \tilde{G}_1 \\ \tilde{G}_2 \\ \vdots \\ \tilde{G}_{N_h} \end{bmatrix}_{V=\hat{V}} + \begin{bmatrix} \partial \tilde{G}_1 / \partial V^T \\ \partial \tilde{G}_2 / \partial V^T \\ \vdots \\ \partial \tilde{G}_{N_h} / \partial V^T \end{bmatrix}_{V=\hat{V}} (V^* - \hat{V}) + O(\hat{V}^T x) \\ &= G_V \tilde{V}^T x + O(\hat{V}^T x) \end{aligned} \quad (38)$$

where:

$$G_V = \left[\frac{\partial G_1}{\partial V^T x}, \frac{\partial G_2}{\partial V^T x}, \dots, \frac{\partial G_{N_h}}{\partial V^T x} \right]^T \in \mathbb{R}^{N_h \times N_h}$$

$$\tilde{V} = V^* - \hat{V} \in \mathbb{R}^{4 \times N_h}$$

$O(\hat{V}^T x) \in \mathbb{R}^{N_h}$ is a vector of higher-order terms and it is assumed to be bounded.

By substituting (38) into the approximation error equation (37), we have:

$$\begin{aligned} \Delta \tau_a - f_{NN} &= W^{*T} G_V \tilde{V}^T x + \tilde{W}^T \hat{G} + W^{*T} O(\hat{V}^T x) + \varepsilon_N \\ &= \hat{W}^T G_V \tilde{V}^T x + \tilde{W}^T \hat{G} + \delta \end{aligned} \quad (39)$$

where:

$$\delta = \tilde{W}^T G_V \tilde{V}^T x + W^{*T} O(\hat{V}^T x) + \varepsilon_N \quad (40)$$

Next, we design an error estimator f_{est} to estimate and compensate for the error vector $\delta \in \mathbb{R}^2$. The online learning algorithms of the NN and the structure of the error estimator f_{est} are derived in the next section following the Lyapunov method.

4.3 Stability analysis

According to the Lyapunov stability analysis, the system is stable if the Lyapunov function candidate is positive definite and its derivative is negative semidefinite.

Consider the Lyapunov function candidate:

$$V = V_1 + V_2 + V_3 + V_4 \quad (41)$$

where:

$$V_1 = \frac{1}{2} s^T \hat{M}_a s \quad (42)$$

$$V_2 = \frac{1}{2} \text{tr}(\tilde{W}^T \eta^{-1} \tilde{W}) \quad (43)$$

$$V_3 = \frac{1}{2} \text{tr}(\tilde{V}^T \mu^{-1} \tilde{V}) \quad (44)$$

$$V_4 = \frac{1}{2} \tilde{\delta}^T \xi^{-1} \tilde{\delta} \quad (45)$$

in which η and μ are the positive learning rates; ξ is a diagonal positive constant matrix of the error estimator (56); and $\tilde{\delta} = \delta - f_{est}$ is the estimated error.

Obviously, V_1 , V_2 , V_3 and V_4 are positive definite functions. Therefore, V is a positive definite function.

The derivative of V_1 is computed as follows:

$$\dot{V}_1 = \frac{1}{2} \left(\dot{s}^T \hat{M}_a s + s^T \dot{\hat{M}}_a s + s^T \hat{M}_a \dot{s} \right) \quad (46)$$

Properties 1 and 2 in section 2 give us:

$$\dot{s}^T \hat{M}_a s = s^T \hat{M}_a \dot{s} \quad (47)$$

$$s^T \dot{\hat{M}}_a s = 2s^T \hat{C}_a s \quad (48)$$

By substituting (47) and (48) into (46) we obtain:

$$\dot{V}_1 = s^T \left[\hat{C}_a s + \dot{\hat{M}}_a s \right]$$

$$= s^T \left[\hat{C}_a s + \hat{M}_a \ddot{\theta}_a - \hat{M}_a \ddot{\theta}_{ar} \right]$$

$$= s^T \left[\hat{C}_a s + \tau_a - \hat{C}_a \dot{\theta}_a - \Delta \tau_a - \hat{M}_a \ddot{\theta}_{ar} \right] \quad (49)$$

Now, substituting the proposed controller (35) into (49) we obtain:

$$\dot{V}_1 = s^T \left[f_{NN} - \Delta \tau_a + f_{est} - Ts \right] \quad (50)$$

From (39) and (50), we have:

$$\dot{V}_1 = s^T \left[-\hat{W}^T G_V \tilde{V}^T x - \tilde{W}^T \hat{G} - \delta + f_{est} - Ts \right] \quad (51)$$

From (41) and (51), we have the derivative of the Lyapunov candidate function:

$$\begin{aligned} \dot{V} &= \dot{V}_1 + \dot{V}_2 + \dot{V}_3 + \dot{V}_4 \\ &= s^T \left[-\hat{W}^T G_V \tilde{V}^T x - \tilde{W}^T \hat{G} - \tilde{\delta} - Ts \right] - \\ &\quad - \text{tr} \left(\tilde{W}^T \eta^{-1} \dot{\tilde{W}} \right) - \text{tr} \left(\tilde{V}^T \mu^{-1} \dot{\tilde{V}} \right) - \tilde{\delta}^T \xi^{-1} \dot{\tilde{\delta}} \end{aligned} \quad (52)$$

As \dot{V} is desired to be at least negative semidefinite, we choose the online update laws for the NN and design the estimator as follows:

$$\dot{\hat{W}} = -\eta \hat{G} s^T \quad (53)$$

$$\dot{\hat{V}} = -\mu \chi \left(G_v \hat{W} s \right)^T \quad (54)$$

$$\dot{f}_{est} = -\xi s \quad (55)$$

$$f_{est} = -\xi \int s dt + \Gamma \quad (56)$$

where equation (56) is derived from equation (55). Moreover, the constant matrix ξ is eliminated from the integration in (56) since the recursive estimation algorithm can recover it. The constant vector Γ is chosen as zero.

Next, by substituting (53)-(56) into (52), we have:

$$\begin{aligned}\dot{V} &= s^T \left[-\hat{W}^T G_V \tilde{V}^T x - \hat{W}^T \hat{G} - \tilde{\delta} - Ts \right] - \\ &+ tr \left(\tilde{W}^T \hat{G} s^T \right) + tr \left(\tilde{V}^T x \left(G_v \hat{W} s \right)^T \right) + \tilde{\delta}^T s \\ &= s^T \left[-\hat{W}^T G_V \tilde{V}^T x - \hat{W}^T \hat{G} - \tilde{\delta} - Ts \right] - \\ &+ s^T \tilde{W}^T \hat{G} + s^T \hat{W}^T G_V \tilde{V}^T x + \tilde{\delta}^T s \\ &= -s^T Ts \leq 0\end{aligned}\quad (57)$$

In (57), since T is a diagonal positive definite matrix, $\dot{V} = 0$ only when $s = 0$. Therefore, we can see that the control system is asymptotically stable with respect to s . This means that:

$$\lim_{t \rightarrow \infty} s = \lim_{t \rightarrow \infty} (\dot{e} + \Lambda e) = 0 \quad (58)$$

Or, equivalently:

$$\begin{cases} \lim_{t \rightarrow \infty} e = 0 \Rightarrow \lim_{t \rightarrow \infty} \theta_a = \theta_{da} \\ \lim_{t \rightarrow \infty} \dot{e} = 0 \Rightarrow \lim_{t \rightarrow \infty} \dot{\theta}_a = \dot{\theta}_{da} \end{cases} \quad (59)$$

Thus, it is proved that, with the proposed controller (35), the actual active joint positions converge on their desired values.

5. Simulation study

Simulation studies were conducted on Matlab-Simulink and the mechanical model of the 2-DOF planar parallel manipulator was built in SimMechanics toolbox. The link parameters in the mechanical model are set as follows: $l_{11} = l_{21} = 0.102(\text{m})$, $l_{12} = l_{22} = 0.18(\text{m})$ and $l_0 = 0.132(\text{m})$ are the link lengths; $r_{11} = 0.05(\text{m})$, $r_{21} = 0.055(\text{m})$, $r_{12} = r_{22} = 0.09(\text{m})$ are the distances from the joint to the centre of mass of the links; $m_{11} = 0.8(\text{kg})$, $m_{21} = 0.78(\text{kg})$, $m_{12} = 1.17(\text{kg})$, $m_{22} = 1.2(\text{kg})$ are the masses of the links; $I_{z11} = 0.0027(\text{kg.m}^2)$, $I_{z21} = 0.0031(\text{kg.m}^2)$, $I_{z12} = I_{z22} = 0.0013(\text{kg.m}^2)$ are the inertia tensors of the links.

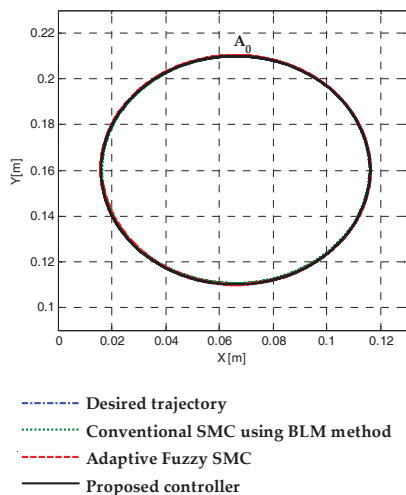


Figure 5. Results of circular trajectory tracking control in case 1.

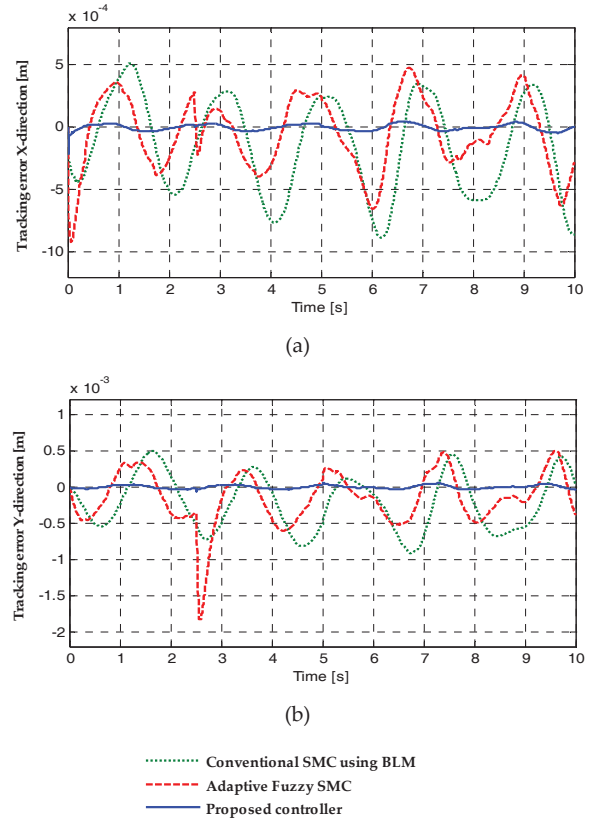


Figure 6. Tracking errors of the end-effector in case 1: (a) X-direction and (b) Y-direction.

In practice, it is difficult to measure the distances from the joint to the centre of mass and the inertia tensors of the links. As such, we conducted the simulations with different parameters, both in the mechanical model of the robot and in the controllers, as follows:

$$\hat{r}_{i1} = 0.9r_{i1} \text{ and } \hat{r}_{i2} = 0.9r_{i2}, \quad i = 1, 2 \quad (60)$$

where \hat{r}_{i1} and \hat{r}_{i2} were used for calculating \hat{M}_a and \hat{C}_a in the controllers.

The friction models of the system, including the viscous friction and the Coulomb friction torques, are defined as follows:

$$f_{ai} = F_{ci} \text{sign}(\dot{\theta}_{ai}) + F_{vi} \dot{\theta}_{ai}, \quad i = 1, 2 \quad (61)$$

where the coefficients are chosen as $F_{c1} = F_{c2} = 0.3$ and $F_{v1} = F_{v2} = 2$.

The simulations were carried out with respect to those situations when the end-effector of the parallel manipulator is driven to track a circular trajectory on the XY plane under different initial conditions. To illustrate the improvement in performance, the proposed controller (35) is compared with two other controllers:

+ A conventional SMC using the boundary layer method (BLM) which was proposed in [14]:

$$\tau_a = \hat{M}_a \ddot{\theta}_{ar} + \hat{C}_a \dot{\theta}_{ar} - K \text{sat}(s/\Phi) \quad (62)$$

where K is a diagonal positive matrix of switching gains chosen as $K = \text{diag}\{10, 10\}$; $\text{sat}(s/\Phi) = [\text{sat}(s_1/\Phi_1), \text{sat}(s_2/\Phi_2)]^T$ is a vector of saturation functions which is defined in [14]; and Φ_1 and Φ_2 are the boundary layer thicknesses, chosen as $\Phi_1 = \Phi_2 = 0.15$.

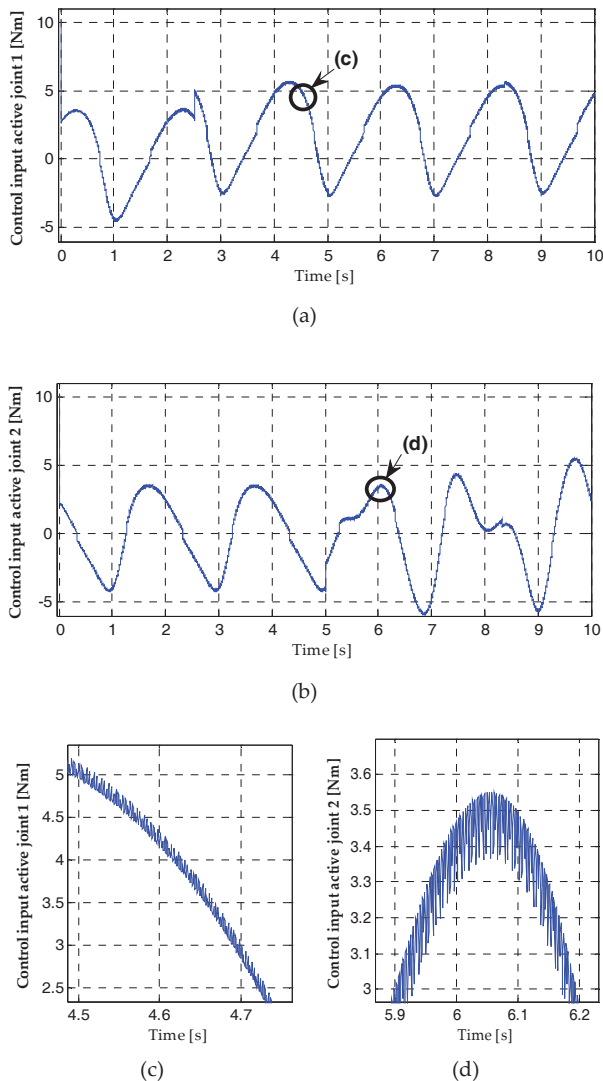


Figure 7. Case 1 when applying the conventional SMC using BLM: (a) Control input of the active joint 1; (b) Control input of the active joint 2; (c) Enlargement of the localized region in (a); (d) Enlargement of the localized region in (b).

+ An adaptive fuzzy sliding mode controller which was proposed in [22]:

$$\tau_a = \hat{M}_a \ddot{\theta}_{ar} + \hat{C}_a \dot{\theta}_{ar} - A s - K_{fuzzy} \quad (63)$$

where A is the diagonal positive parameter matrix chosen as $A = \text{diag}\{10, 10\}$; and $K_{fuzzy} = [k_{fuzzy1}, k_{fuzzy2}]^T$ in which each k_{fuzzyi} is estimated by an individual SISO fuzzy system ($i = 1, 2$).

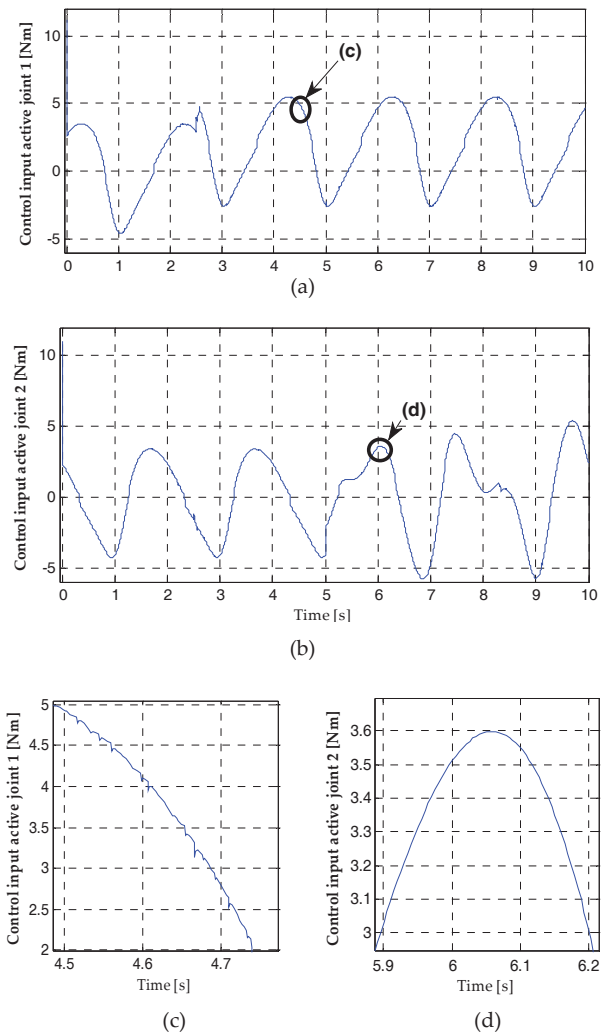


Figure 8. Case 1 when applying the adaptive fuzzy SMC: (a) Control input of the active joint 1; (b) Control input of the active joint 2; (c) Enlargement of the localized region in (a); (d) Enlargement of the localized region in (b).

The parameters in the proposed controller are set as: $T = \text{diag}\{10, 10\}$; the number of neurons in the hidden layer of the NN is 10; the learning rates are chosen as $\mu = 10^{-3}$ and $\eta = 2 \times 10^{-3}$; the initial weight matrices are $W(0) = 0.001 \times I_{4 \times 10}$ and $V(0) = 0.001 \times I_{10 \times 2}$; the constant matrix in the error estimator (50) is chosen as $\xi = \text{diag}\{5000, 5000\}$.

In simulations, the 2-DOF parallel manipulator is disturbed by step external disturbance forces $d_{a1}(t) = [2, 0]^T$ at $t = 2.5$ s (on active joint 1) and periodic external disturbance forces $d_{a2}(t) = [0, 2\sin(1.7\pi t)]^T$ at $t = 5$ s (on active joint 2).

First, in case 1, we conduct the simulation when the initial position of the end-effector $E(x, y)$ of the parallel manipulator lies on the top of the reference circular trajectory $A_0(0.066, 0.21)$. The centre coordinates of the reference circular trajectory are $(0.066, 0.16)$ and the radius is 0.05. The end-effector is driven to track the

circular trajectory 5 times over 10 seconds. Figure 5 shows the results of the tracking control of the 2-DOF parallel manipulator in case 1.

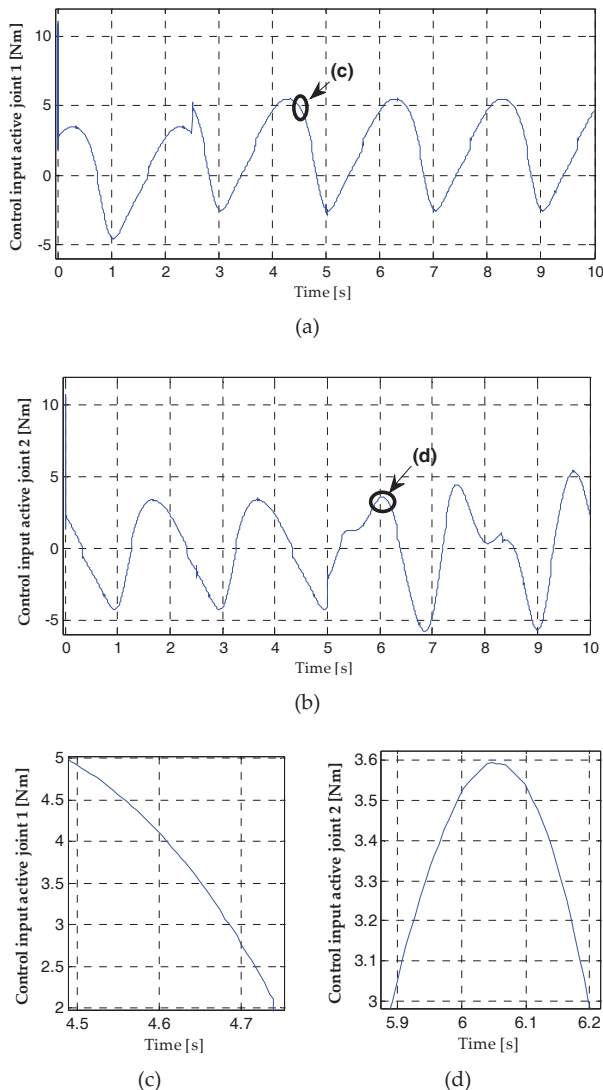


Figure 9. Case 1 when applying the proposed controller: (a) Control input of the active joint 1; (b) Control input of the active joint 2; (c) Enlargement of the localized region in (a); (d) Enlargement of the localized region in (b).

Figure 6 shows the tracking errors of the end-effector in the X-direction and in the Y-direction in case 1. As can be seen from the figure, the tracking errors caused by the adaptive fuzzy SMC are a little bit smaller than the errors associated with the conventional SMC using BLM. In particular, the proposed controller brings about the smallest tracking errors (almost converging on zero) compared with the conventional SMC using BLM and the adaptive fuzzy SMC. In addition, it can be seen that the large model uncertainties and external disturbances are greatly compensated for using the proposed controller.

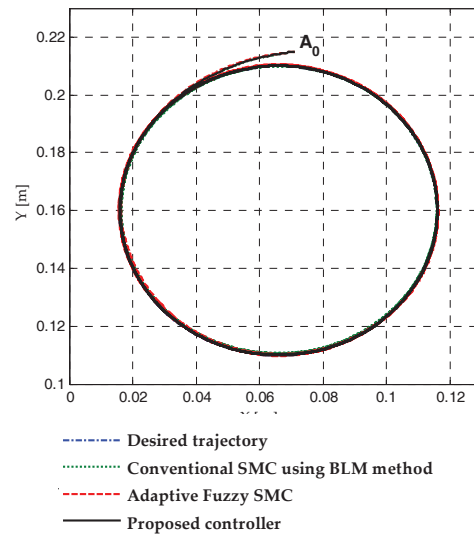


Figure 10. Results of the circular trajectory tracking control in case 2.

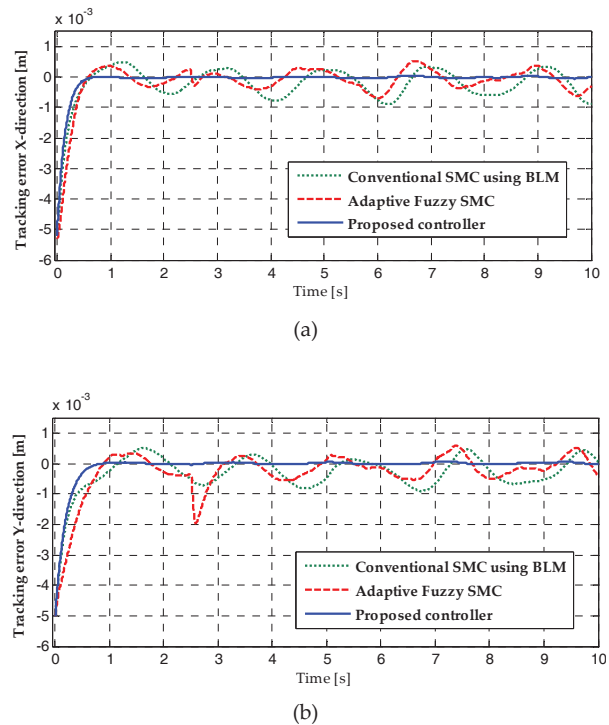


Figure 11. Tracking errors of the end-effector in case 2: (a) X-direction and (b) Y-direction.

The control inputs of the active joints 1 and 2 in case 1 of the conventional SMC using BLM are shown in Figure 7. From the enlargements of the localized regions, it can clearly be seen that the chattering phenomenon remains. If we increase the boundary layer thickness or decrease the switching gains for reducing greater reduction of the chattering, the tracking errors will be increased and the robustness of the system will not be guaranteed.

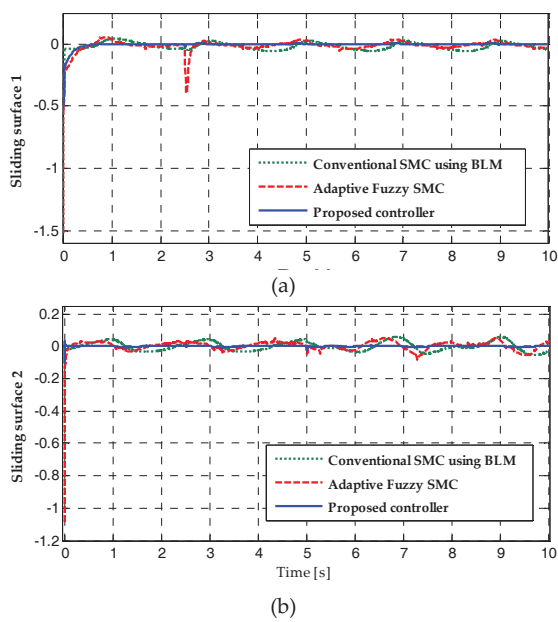


Figure 12. Sliding surfaces of (a) active joint 1 and (b) active joint 2 in case 2.

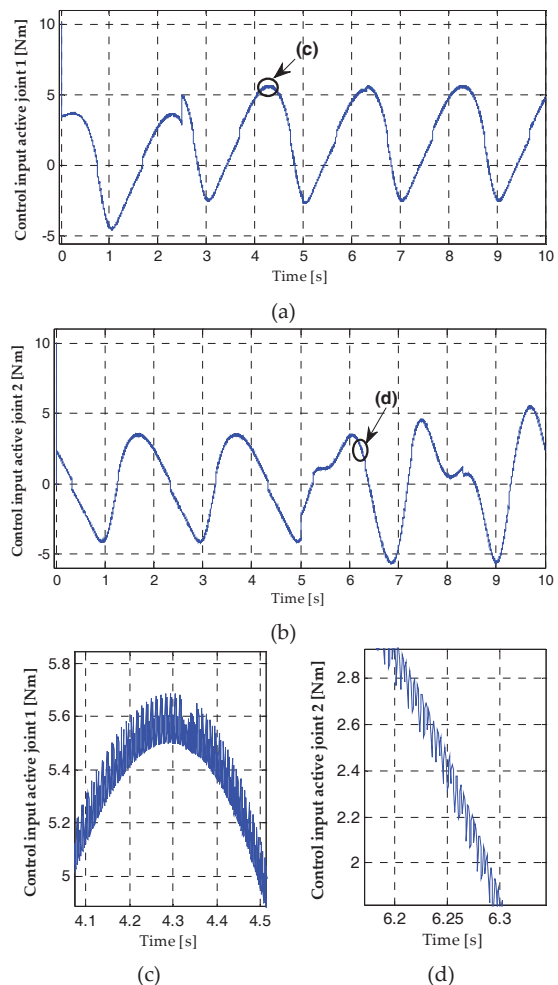


Figure 13. Case 2 with the conventional SMC using BLM: (a) Control input of the active joint 1; (b) Control input of the active joint 2; (c) Enlargement of the localized region in (a); (d) Enlargement of the localized region in (b).

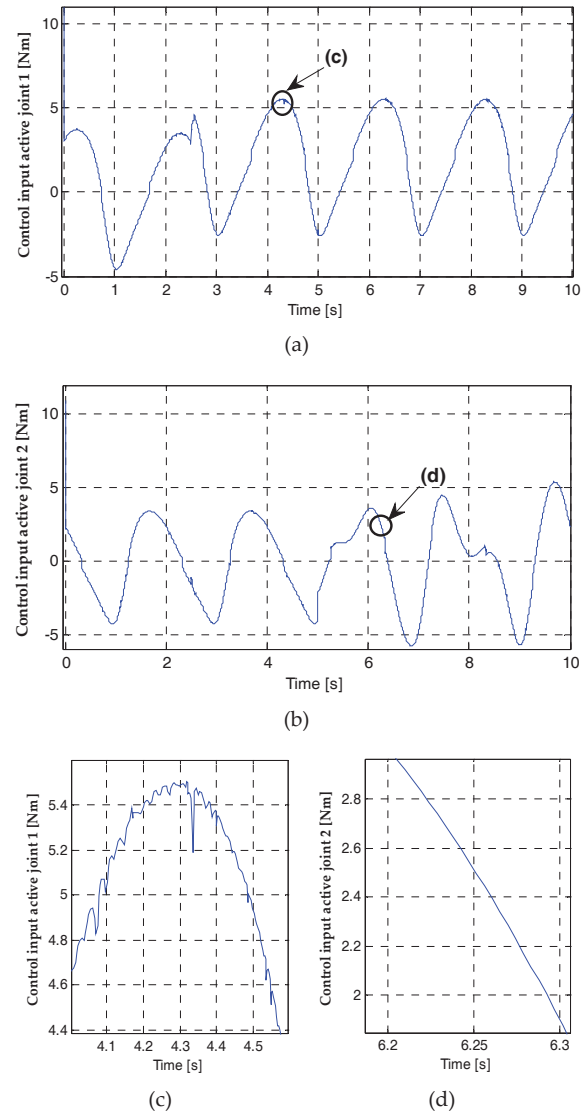


Figure 14. Case 2 with the adaptive fuzzy SMC: (a) Control input of the active joint 1; (b) Control input of the active joint 2; (c) Enlargement of the localized region in (a); (d) Enlargement of the localized region in (b).

Figures 8 and 9 show the control inputs of the active joints 1 and 2 in case 1 using the adaptive fuzzy SMC and the proposed controller, respectively. Compared with the conventional SMC using BLM, it can be seen that the chattering is removed. However, the proposed controller can completely compensate for the uncertainties and any external disturbances; hence, the tracking errors in the case of using the proposed controller are almost reduced to zero, as shown in Figure 6.

Next, a simulation is carried out to investigate the control performance when the end-effector $E(x,y)$ of the parallel manipulator does not lie on the reference circle. In this case, we can demonstrate the convergence of the tracking errors and sliding functions. Figure 10 shows the comparison of the trajectory tracking among the conventional SMC using BLM, the adaptive fuzzy SMC and the proposed controller. The centre coordinates of the

reference circular trajectory are (0.066, 0.16) and the radius is 0.05. The initial position of the end-effector $E(x,y)$ of the parallel manipulator is A_0 (0.071,0.215). The values of the parameters in the controllers for case 2 are set to be the same as with case 1 of the simulations.

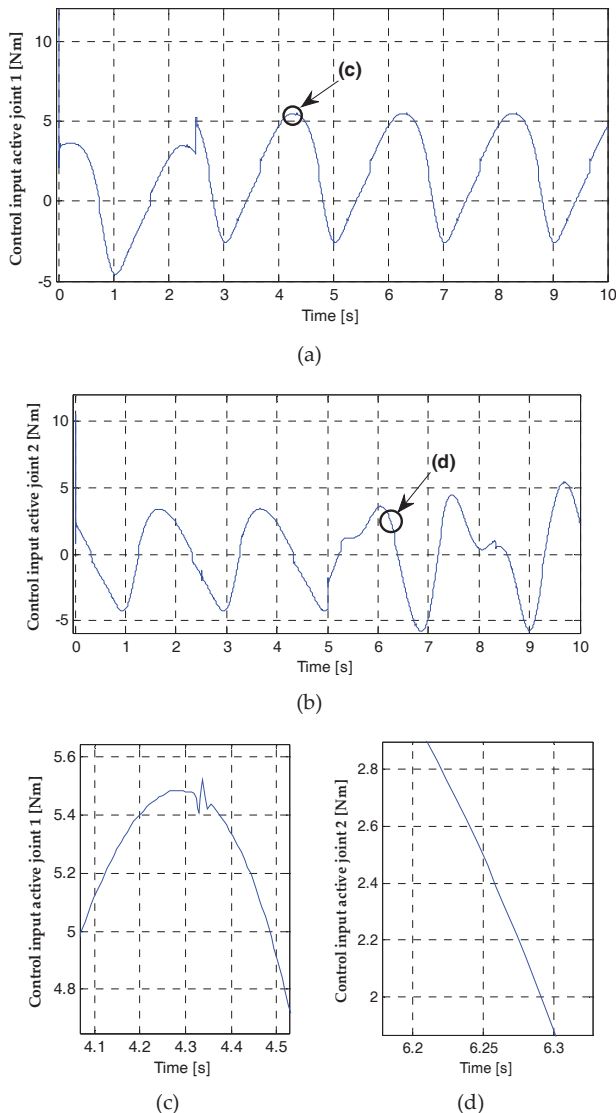


Figure 15. Case 2 with the proposed controller: (a) Control input of the active joint 1; (b) Control input of the active joint 2; (c) Enlargement of the localized region in (a); (d) Enlargement of the localized region in (b).

The results of the tracking errors of the end-effector in case 2 on the X-direction and the Y-direction - which are shown in Figure 11 - prove the excellence of the control performance of the proposed controller in comparison with the conventional SMC using BLM and the adaptive fuzzy SMC. It can be seen that the proposed controller brings about the smallest and the fastest convergence of tracking errors.

Figure 12 shows the comparison of the convergence of the sliding functions among all three controllers. As can be seen from the figure, the sliding functions in the case

using the proposed controller converge on the smallest values (almost equal to zero).

Figures 13-15 show the control inputs of the active joints 1 and 2 in case 2 for the conventional SMC using BLM, the adaptive fuzzy SMC and the proposed controller, respectively. From the enlargements of the localized regions, it can clearly be seen that the chattering phenomenon remains in the case of the conventional SMC using BLM but that it is eliminated in the cases of the adaptive fuzzy SMC and the proposed controller.

It can be concluded from the above-mentioned simulation results that the proposed controller is highly efficient in the control of the 2-DOF planar parallel manipulators.

6. Conclusions

In this paper, we have presented a novel chattering-free neuron sliding mode controller for tracking the control of 2-DOF parallel manipulators. The proposed controller is based on the combination of three ingredients. The first ingredient is the equivalent control, which is derived from the dynamic model of the 2-DOF parallel manipulators. The second one is a feedforward NN used to adaptively learn the large nonlinear uncertainties and external disturbances of the parallel manipulators. The final part is an error estimator for compensating the approximation errors of the NN and the higher-order terms in Taylor series expansion. The online weight tuning algorithms of the NN and the structure of the error estimator are derived with the strict theoretical stability proof of the Lyapunov theorem. The connection weights of the NN can be adapted online during the trajectory tracking control of the parallel manipulators without any offline training phase. The main advantages of the proposed controller in comparison with the existing SMC methods are as follows: (1) The proposed controller can completely compensate the large nonlinear uncertainties and external disturbances of parallel manipulators. (2) The proposed control strategy does not need to know either the upper bounds of any uncertainties or the bound of any approximation errors. Its structure is simple, easy to implement and yet leads to the acquisition of good results. These advantages mean that the proposed controller is suitable in application to those tracking control parallel manipulators which have a complicated dynamic model and huge uncertainties.

Simulation results demonstrated the effectiveness of the proposed controller in the trajectory tracking control of a 2-DOF parallel manipulator. It has been shown that the proposed controller brings about the smallest tracking errors compared with the conventional SMC using BLM and the adaptive fuzzy SMC. The chattering in the control inputs is eliminated and the control system is highly robust against uncertainties and external disturbances.

7. Acknowledgements

The authors would like to express their thanks for financial support from the Korean Ministry of Knowledge Economy under the Human Resources Development Program for Convergence Robot Specialists. (NIPA-2012-H1502-12-1002)

8. References

- [1] Merlet J. P. (2006) Parallel Robots. Second ed.: Springer.
- [2] Ghorbel F. H., Chetelat O., Gunawardana R. and Longchamp R. (2000) Modeling and set point control of closed-chain mechanisms: theory and experiment. Control Systems Technology, IEEE Transactions on, vol. 8, pp. 801-815.
- [3] Ouyang P. R., Zhang W. J. and Wu F. X. (2002) Nonlinear PD control for trajectory tracking with consideration of the design for control methodology. Robotics and Automation, Proceedings. ICRA '02. IEEE International Conference on, pp. 4126-4131 .
- [4] Ouyang P. R., Zhang W. J. and Gupta M. M. (2006) An adaptive switching learning control method for trajectory tracking of robot manipulators. Mechatronics, vol. 16, pp. 51-61.
- [5] Lu R., Mills J. K. and Dong S. (2008) Trajectory Tracking Control for a 3-DOF Planar Parallel Manipulator Using the Convex Synchronized Control Method. Control Systems Technology, IEEE Transactions on, vol. 16, pp. 613-623.
- [6] Yuxin S., Dong S., Lu R. and Mills J. K. (2006) Integration of saturated PI synchronous control and PD feedback for control of parallel manipulators. Robotics, IEEE Transactions on, vol. 22, pp. 202-207.
- [7] Weiwei S., Shuang C., Yaixin Z. and Yanyang L. (2009) Active Joint Synchronization Control for a 2-DOF Redundantly Actuated Parallel Manipulator. Control Systems Technology, IEEE Transactions on, vol. 17, pp. 416-423.
- [8] Yang Z., Wu J. and Mei J. (2007) Motor-mechanism dynamic model based neural network optimized computed torque control of a high speed parallel manipulator. Mechatronics, vol. 17, pp. 381-390.
- [9] Paccot F., Andreff N. and Martinet P. (2009) A Review on the Dynamic Control of Parallel Kinematic Machines: Theory and Experiments. The International Journal of Robotics Research, vol. 28, pp. 395-416.
- [10] Müller A. and Hufnagel T. (2012) Model-based control of redundantly actuated parallel manipulators in redundant coordinates. Robotics and Autonomous Systems, vol. 60, pp. 563-571.
- [11] Le T. D., Kang H.-J., Suh Y.-S. and Ro Y.-S. (2012) An Online Self Gain Tuning Method Using Neural Networks for Nonlinear PD Computed Torque Controller of a 2-dof Parallel Manipulator, Neurocomputing, <http://dx.doi.org/10.1016/j.neucom.2012.01.047>.
- [12] Abdellatif H. and Heimann B. (2010) Advanced Model-Based Control of a 6-DOF Hexapod Robot: A Case Study. Mechatronics, IEEE/ASME Transactions on, vol. 15, pp. 269-279.
- [13] Xiaocong Z., Guoliang T., Bin Y. and Jian C. (2009) Integrated Direct/Indirect Adaptive Robust Posture Trajectory Tracking Control of a Parallel Manipulator Driven by Pneumatic Muscles. Control Systems Technology, IEEE Transactions on, vol. 17, pp. 576-588.
- [14] Slotine J. J. E. and Li W. (1991) *Applied Nonlinear Control*: Prentice-Hall.
- [15] Young K. D., Utkin V. I. and Ozguner U. (1999) A control engineer's guide to sliding mode control. Control Systems Technology, IEEE Transactions on, vol. 7, pp. 328-342.
- [16] Sadati N. and Ghadami R. (2008) Adaptive multi-model sliding mode control of robotic manipulators using soft computing. Neurocomputing, vol. 71, pp. 2702-2710.
- [17] Zeinali M. and Notash L. (2010) Adaptive sliding mode control with uncertainty estimator for robot manipulators. Mechanism and Machine Theory, vol. 45, pp. 80-90.
- [18] Kim N.-I., Lee C.-W. and Chang P.-H. (1998) Sliding mode control with perturbation estimation: application to motion control of parallel manipulator. Control Engineering Practice, vol. 6, pp. 1321-1330.
- [19] Qi Z., McInroy J. and Jafari F. (2007) Trajectory Tracking with Parallel Robots Using Low Chattering, Fuzzy Sliding Mode Controller. Journal of Intelligent & Robotic Systems, vol. 48, pp. 333-356.
- [20] Yang Z., Huang T., Xu X. and Cooper J. (2007) Variable structure control of high-speed parallel manipulator considering the mechatronics coupling model. The International Journal of Advanced Manufacturing Technology, vol. 34, pp. 1037-1051.
- [21] Shang W.-w., S. Cong and S.-I. Jiang (2010) Dynamic model based nonlinear tracking control of a planar parallel manipulator. Nonlinear Dynamics, vol. 60, pp. 597-606.
- [22] Yuzheng G. and Peng-Yung W. (2003) An adaptive fuzzy sliding mode controller for robotic manipulators. Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on, vol. 33, pp. 149-159.
- [23] Byung Kook Y. and Woon Chul H. (2000) Adaptive control of robot manipulator using fuzzy compensator. Fuzzy Systems, IEEE Transactions on, vol. 8, pp. 186-199.
- [24] Chih-Lyang H. and Chia-Ying K. (2001) A stable adaptive fuzzy sliding-mode control for affine nonlinear systems with application to four-bar linkage systems. Fuzzy Systems, IEEE Transactions on, vol. 9, pp. 238-252.

- [25] Ciliz M. K. (2005) Adaptive control of robot manipulators with neural network based compensation of frictional uncertainties. *Robotica*, vol. 23, pp. 159-167.
- [26] Ertugrul M. and Kaynak O. (2000) Neuro sliding mode control of robotic manipulators. *Mechatronics*, vol. 10, pp. 239-263.
- [27] Murray R. M., Zexiang L. and Sastry S. S. (1994) *A Mathematical Introduction to Robotic Manipulation*. CRC Press, Boca Raton.
- [28] Nakamura Y. and Yamane K. (2000) Dynamics computation of structure-varying kinematic chains and its application to human figures. *Robotics and Automation, IEEE Transactions on*, vol. 16, pp. 124-134.
- [29] Hui C., Yiu-Kuen Y. and Zexiang L. (2003) Dynamics and control of redundantly actuated parallel manipulators. *Mechatronics, IEEE/ASME Transactions on*, vol. 8, pp. 483-491.