Fuzzy Adaptive Synchronized Sliding Mode Control Of Parallel Manipulators

Tien Dung Le
The University of Danang - University of Science and Technology
54 Nguyen Luong Bang street, Danang, Vietnam
Email: ltdung@dut.udn.vn

Quang Vinh Doan
The University of Danang - University of Science and Technology
54 Nguyen Luong Bang street, Danang, Vietnam
Email: dqvinh@dut.udn.vn

ABSTRACT
Control of parallel manipulators is a challenging and difficult task. In this paper, we propose a new adaptive synchronized sliding mode controller for planar parallel manipulators by using fuzzy logic system. The proposed controller is based on the combination of the definition of synchronized control, sliding mode control and online self-tuning fuzzy logic system. Firstly, the dynamic model of parallel manipulators is presented in active joint space. Based on this dynamic model, a synchronized sliding mode controller is developed. And then, in order to compensate the uncertainties of the control system, an online self-tuned fuzzy logic system is adopted. The self-tuning law is proposed by using the cross-coupling error of the parallel manipulator. The stability of the closed loop system is guaranteed by using Lyapunov theory. The simulations were conducted on Matlab/Simulink to verify the effectively of the proposed control algorithm.

CCS Concepts
• **Computing methodologies → Artificial intelligence → Control methods**

Keywords
Parallel robotic manipulators; Synchronized control algorithm; Fuzzy logic system; Online self-tuning; Cross-coupling error

1. INTRODUCTION
Parallel manipulators are increasingly in demand for use in many applications such as flight simulators, automobile simulators, work processes, assembly of PCBs, humanoid robots, medical robot, etc. These applications require the potential advantages of parallel manipulators such as high speed, high stiffness, high accuracy, heavy payload capacity, and so on [1].

ICMRE 2018, February 7–11, 2018, Valenciennes, France

ACM ISBN 978-1-4503-6365-5/18/02…$15.00
DOI: https://doi.org/10.1145/3191477.3199060

However, parallel manipulators have small workspaces, abundant singularities, and difficult kinematics, which make the control of parallel manipulators become a challenging and difficult task.

How to develop an effective control method for precise tracking control of parallel manipulators attracts much attention from academe and industry. In most of existing approaches [2-5], the proposed control algorithms are adopted from the conventional methods which were used for serial manipulators. These control approaches did not consider the synchronization among control loops. The lack of synchronization will lead to large coupling errors and degrade the performance of overall system [6]. Recently, the synchronized control methods were proposed to overcome this problem. The synchronized control approach was initially proposed by Koren [7] and applied to motion control for multi-axis machine tools. This control approach is then applied to control many types of mechanical systems such as parallel robotic manipulators, multiple mobile robots, co-operative robotic manipulators, and multi-agent systems. For each type of these systems, the kinematics and dynamic model should be considered. The synchronized control algorithm is designed based on three main concepts: The synchronization function, the cross-coupling error, and the control law (such as sliding mode control, adaptive control, computed torque control, etc).

For parallel robotic manipulators, there are some synchronized tracking control approaches were developed. In [8], Y. Su et al proposed the integrated saturated PI synchronous control combined with PD feedback control approach. On the other hand, D. Sun et al [9] presented a feedforward compensation plus cross-coupling error feedback control approach. In addition, other researchers proposed the model-based synchronous control approaches such as adaptive synchronized control [10,11], synchronized computed torque control [12], sliding mode control [13], and so on. The model-based synchronized control approaches bring about the better performance than the error-based synchronized control approach. However, because of the complex in kinematic structure, it is difficult to have the exact dynamic model of the parallel manipulators.

In this paper, by using fuzzy logic system, we propose an adaptive synchronized sliding mode control algorithm for planar parallel robotic manipulators. The proposed controller is based on the combination of the definition of synchronized control, sliding mode control and online self-tuning fuzzy logic system. The dynamic model of parallel manipulators is presented in active joint space. Based on this dynamic model, a synchronized sliding mode controller is developed. And then, an online self-tuned fuzzy logic system is adopted to compensate the uncertainties of the control system.

The rest of the paper is organized as the following. In section 2, the dynamic model of parallel manipulators is presented in active joint space. The proposed control algorithm is described in section 3. In section 4, the simulation studies for a 3-DOF planar parallel robotic manipulator are conducted and compare with other conventional algorithms to verify the effectiveness of the propose algorithm. Finally, some concluding remarks are presented in Section 5.

2. DYNAMIC MODEL

We consider a general parallel manipulator consisting of a number of serial kinematic chains formed by rigid links and joints as depicted in Figure 1a.

Let N_a be the number of active joints, $\theta_a \in R^{Na}$ the actuated joints vector, $\tau_a \in R^{Na}$ the actuated torques vector, N_p the number of passive joints, $\theta_p \in R^{Np}$ the passive joints vector. The active joints are actuated by actuators while the passive joints are free to move. We need to derive the dynamic model of robot in the active joint space as follows.

First, several passive joints are virtually cut to form an equivalent open-chain system. There are at least two possible ways to peform this task as depicted in Figure 1b1 and Figure 1b2. Let N_O be the number of joints in the equivalent open-chain system, $\theta_O \in R^{No}$ the joint angles vector, and $\tau_O \in R^{No}$ the joint toques vector.

Second, it is assumed that $\tau_O \in R^{No}$ generated for a given motion satisfies the loop constraints of the original parallel manipulator, and there is no force or moment interaction at virtually cut joints. By using Newton-Euler or Lagrangian approaches, dynamic model of the equivalent open-chain system can be computed and given as the following equation:

$$\boldsymbol{M}_O\ddot{\boldsymbol{\theta}}_O + \boldsymbol{C}_O\dot{\boldsymbol{\theta}}_O + \boldsymbol{G}_O + \boldsymbol{F}_O = \boldsymbol{\tau}_O \tag{1}$$

where $\boldsymbol{M}_O \in \mathbf{R}^{No\times No}$ is the inertia matrix; $\boldsymbol{C}_O \in \mathbf{R}^{No\times No}$ is the Coriolis and centrifugal force matrix; $\boldsymbol{G}_O \in \mathbf{R}^{No}$ is the gravity force vector; and $\boldsymbol{F}_O \in \mathbf{R}^{No}$ is the friction force vector.

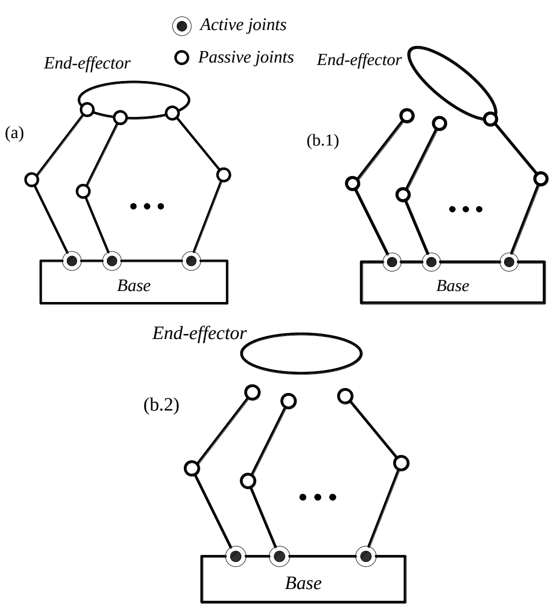

Figure 1. A typical parallel manipulator and its equivalent open-chain systems

Following the D'Alembert principle and principle of virtual work, the joint torque vectors τ_O and τ_a satisfy the following equation [14]:

$$\boldsymbol{\tau}_a = \boldsymbol{\Psi}^T\boldsymbol{\tau}_O \tag{2}$$

where $\Psi = \partial\theta_O/\partial\theta_a \in R^{No\times Na}$ is the Jacobian matrix given by the loop constraints.

In addition, we have the following relationships:

$$\dot{\boldsymbol{\theta}}_O = \frac{\partial\boldsymbol{\theta}_O}{\partial\boldsymbol{\theta}_a}\dot{\boldsymbol{\theta}}_a = \boldsymbol{\Psi}\dot{\boldsymbol{\theta}}_a \text{ and } \ddot{\boldsymbol{\theta}}_O = \dot{\boldsymbol{\Psi}}\dot{\boldsymbol{\theta}}_a + \boldsymbol{\Psi}\ddot{\boldsymbol{\theta}}_a \tag{3}$$

By multiplying both sides of (1) with Ψ^T, and then substituting (2) and (3) into the new equation we obtain:

$$\boldsymbol{\Psi}^T\boldsymbol{M}_O\boldsymbol{\Psi}\ddot{\boldsymbol{\theta}}_a + \left(\boldsymbol{\Psi}^T\boldsymbol{M}_O\dot{\boldsymbol{\Psi}} + \boldsymbol{\Psi}^T\boldsymbol{C}_O\boldsymbol{\Psi}\right)\dot{\boldsymbol{\theta}}_a + \boldsymbol{\Psi}^T\boldsymbol{G}_O + \boldsymbol{\Psi}^T\boldsymbol{F}_O = \boldsymbol{\tau}_a \tag{4}$$

Finally, the dynamic model of the general parallel manipulators can be expressed by [15]:

$$\hat{\boldsymbol{M}}_a\ddot{\boldsymbol{\theta}}_a + \hat{\boldsymbol{C}}_a\dot{\boldsymbol{\theta}}_a + \hat{\boldsymbol{G}}_a + \boldsymbol{F}_a = \boldsymbol{\tau}_a \tag{5}$$

where $\hat{\boldsymbol{M}}_a = \boldsymbol{\Psi}^T\boldsymbol{M}_O\boldsymbol{\Psi} \in \boldsymbol{R}^{\boldsymbol{Na}\times\boldsymbol{Na}}$ is the estimated inertia matrix,

$\hat{\boldsymbol{C}}_a = \boldsymbol{\Psi}^T\boldsymbol{M}_O\dot{\boldsymbol{\Psi}} + \boldsymbol{\Psi}^T\boldsymbol{C}_O\boldsymbol{\Psi} \in \boldsymbol{R}^{Na\times Na}$ is the estimated Coriolis and centrifugal force matrix,

$\hat{\boldsymbol{G}}_a = \boldsymbol{\Psi}^T\boldsymbol{G}_O \in \boldsymbol{R}^{Na}$ is the estimated gravity force vector, and

$\boldsymbol{F}_a = \boldsymbol{\Psi}^T\boldsymbol{F}_O \in \boldsymbol{R}^{Na}$ is the vector of friction force of the parallel manipulator.

Note that the effect of friction forces and external disturbances on the passive joints is often much smaller than on the active joints. Thus, in order to simplify the dynamic model of parallel manipulators, only those terms on the active joints are considered.

The dynamic model (5) has the following properties [13]:

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

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

In practice, due to the presence of the highly nonlinear uncertainties and external disturbances, the precise dynamic model of parallel manipulators will never be known. Hence we can express the actual dynamic model of general parallel manipulators as follows:

$$\hat{\boldsymbol{M}}_a\ddot{\boldsymbol{\theta}}_a + \hat{\boldsymbol{C}}_a\dot{\boldsymbol{\theta}}_a + \hat{\boldsymbol{G}}_a + \Delta\boldsymbol{\tau}_a = \boldsymbol{\tau}_a \tag{6}$$

where $\Delta\boldsymbol{\tau}_a = \Delta\boldsymbol{M}_a\ddot{\boldsymbol{\theta}}_a + \Delta\boldsymbol{C}_a\dot{\boldsymbol{\theta}}_a + \Delta\boldsymbol{G}_a + \boldsymbol{F}_a + \boldsymbol{d}_a(t)$ is the vector of the dynamic modeling errors, uncertainties and external disturbances of the parallel manipulator; $\Delta\boldsymbol{M}_a$, $\Delta\boldsymbol{C}_a$ and $\Delta\boldsymbol{G}_a$ are the unknown bounded modeling errors; and $\boldsymbol{d}_a(t)$ is the external disturbances vector.

3. PROPOSED CONTROLLER

3.1 Fuzzy Logic System

The fuzzy logic system (FLS) used in this paper is a MIMO-FLS. It consists of four parts: the knowledge base, the fuzzifier, the fuzzy inference engine working on fuzzy rules, and the defuzzifier.

The MIMO If-Then rules are of the following form [16]:

$$R = \bigcup_{k=1}^{M} R_k \tag{7}$$

in which each rule R_k is of the form:

R_k: IF x_1 is A_1^k, x_2 is A_2^k, …, x_n is A_n^k, THEN y_1 is B_1^k ,…, y_m is B_m^k

where M is the total number of rules; $x = [x_1,…,x_n]^T$ and $y = [y_1,…,y_m]^T$ are the input and output vectors of the fuzzy system, respectively. A_i^k and B_j^k are the linguistic variables of the fuzzy sets, described by their membership functions $\mu_{A_i^k}(x_i)$ and $\mu_{B_i^k}(y_i)$,where $k = 1,2,…,M$.

The output of the FLS with center-average defuzzifier, product inference, and singleton fuzzifier are of the following form:

$$y_j = \frac{\sum_{k=1}^{M}\left(\prod_{i=1}^{n}\mu_{A_i^k}(x_i)\right)W_j^k}{\sum_{k=1}^{M}\left(\prod_{i=1}^{n}\mu_{A_i^k}(x_i)\right)}, \quad j = 1,2,...,m \tag{8}$$

where W_j^k is the point at which $\mu_{B_i^k}(y_i)$ achieves its maximum value.

Normally, W_j^k are fixed, nonadaptive. In this paper, we use FLS to approximate the lumped uncertainty of the robot manipulator. So W_j^k in this paper are chosen to be free parameters, and the FLS can be viewed as an adaptive fuzzy system. The output of the adaptive FLS can be rewritten as:

$$y_j = \sum_{k=1}^{M} W_j^k \Phi_k(\boldsymbol{x}) = \boldsymbol{W}_j^T \boldsymbol{\Phi}(\boldsymbol{x}), \quad j = 1,2,...,m \tag{9}$$

in which $\boldsymbol{\Phi}(\boldsymbol{x}) = [\Phi_1(\boldsymbol{x}), …, \Phi_M(\boldsymbol{x})]^T$ is the fuzzy basis function vector, and $\boldsymbol{W}_j^T = [W_j^1,...,W_j^M]^T$ is the parameter vector. In the FLS of this paper, W_j^k are adjustable (k=1,…,M; j=1,…,m).

The equation of the fuzzy basic function is expressed as the following:

$$\Phi_k(x) = \frac{\prod_{i=1}^{n}\mu_{A_i^k}(x_i)}{\sum_{k=1}^{M}\left(\prod_{i=1}^{n}\mu_{A_i^k}(x_i)\right)}, \quad k = 1,2,...,M \tag{10}$$

Hence, the MIMO-FLS can be rewritten as:

$$\boldsymbol{y} = \boldsymbol{W}^T\boldsymbol{\Phi}(\boldsymbol{x}) \tag{11}$$

where W is an ($M \times m$) matrix.

3.2 Fuzzy Adaptive Synchronized Sliding Mode Controller

Difference from conventional control methods, the synchronized control method employs an additional feedback signal termed the synchronization error. The synchronization error is typically designed through the combination of position and velocity errors of all actuated joints with a coefficient matrix [11].

Firstly, the tracking error of the ith active joint is defined as:

$$e_i = \theta_{ai}(t) - \theta_{ai}^d(t) \text{ , } i = 1, 2,.., n \tag{12}$$

in which $\theta_{ai}^d(t)$ is the desired angular, $\theta_{ai}(t)$ is the actual angular of of the ith active joint of the parallel robot manipulator; n is the number of degree-of-freedom of the parallel robot manipulator.

To synchronize the motion of all active joints, the key problem is to maintain some special kinematic relationship among active joints. The synchronization function is written as [17]:

$$f(\theta_{a1},\theta_{a2},...,\theta_{an}) : c_1\theta_{a1} = c_2\theta_{a2} = ... = c_n\theta_{an} \tag{13}$$

where c_i is the positive coupling coefficient of ith active joint. The function (13) is valid for all desired coordinates:

$$f(\theta_{a1}^d,\theta_{a2}^d,...,\theta_{an}^d) : c_1\theta_{a1}^d = c_2\theta_{a2}^d = ... = c_n\theta_{an}^d \tag{14}$$

From (13) and (14), the synchronization goal is defined as the following [17]:

$$c_1e_1 = c_2e_2 = ... = c_ne_n \tag{15}$$

Next, the synchronization error of the parallel manipulators can be defined as follows:

$$\begin{cases} \varepsilon_1 = c_1e_1 - c_2e_2 \\ \varepsilon_2 = c_2e_2 - c_3e_3 \\ \quad\vdots \\ \varepsilon_n = c_ne_n - c_1e_1 \end{cases} \tag{16}$$

in which ε_i (i = 1,..,n) represents the synchronization error of ith active joint.

Based on (15) and (16), the cross-coupling error is defined as [18]:

$$\xi_i = c_1e_1 + \mu\int_0^t\left(\varepsilon_i(\omega) - \varepsilon_{i-1}(\omega)\right)d\omega \tag{17}$$

where μ is a positive coupling parameter, ω is a variable from time zero to t, when i = 1, i-1 = n.

For developing the synchronized sliding mode control algorithm for parallel manipulators, the sliding surface function is designed as the following:

$$\boldsymbol{s}^* = \boldsymbol{\xi} + \Gamma\dot{\boldsymbol{\xi}} = \dot{\boldsymbol{\theta}}_a - \dot{\boldsymbol{\theta}}_a^r \tag{17}$$

in which $\boldsymbol{\Gamma} = diag(\lambda_1, \lambda_2,…, \lambda_n)$, and λ_i (i = 1,…,n) are positive constants which determine the motion feature in the sliding surface; $\dot{\boldsymbol{\theta}}_a^r = \dot{\boldsymbol{\theta}}_a^d - \Delta\boldsymbol{\varepsilon}$ is the reference velocity of active joints.

In general, a sliding mode control algorithm is developed such that the system state trajectories are driven to the sliding surface and kept on the sliding surface. The control input vector of the parallel manipulator consists of two components:

$$\boldsymbol{\tau}_a = \boldsymbol{\tau}_{eq} + \boldsymbol{\tau}_{sw} \tag{18}$$

where the first term $\boldsymbol{\tau}_{eq}$ is the equivalent control which keeps the trajectory of the system states on the sliding surface; and second term $\boldsymbol{\tau}_{sw}$ is the switching control which drives the system states toward the sliding surface when they are deviating from this surface.

The equations of the equivalent control and switching control are written as:

$$\boldsymbol{\tau}_{eq} = \hat{\boldsymbol{M}}_a\ddot{\boldsymbol{\theta}}_a^r + \hat{\boldsymbol{C}}_a\dot{\boldsymbol{\theta}}_a^r + \hat{\boldsymbol{G}}_a \tag{19}$$

$$\boldsymbol{\tau}_{sw} = -\boldsymbol{\Lambda}\boldsymbol{s}^* - \boldsymbol{K}sign(\boldsymbol{s}^*) \tag{20}$$

where $\boldsymbol{K} = diag(k_1, k_2,…, k_n)$; $k_1, k_2,…, k_n$ are positive constants; $sign$(s) is the signum function of the sliding surface; and $\boldsymbol{\Lambda} = diag[a_1, a_2,…,a_n]$ is also a diagonal positive definite matrix in which a_i is a positive constant.

The *sign* function in discontinuous control term (21) leads to high-frequency control switching and chattering across the sliding surface. The most common method so called "*boundary layer method*" (BLM) was used for reducing the chattering phenomenon. However, there is a tradeoff between asymptotic tracking and chattering elimination for the width of the boundary layer. A thicker boundary layer would reduce the chattering but make the tracking error bigger, and vice versa.

In this paper, we use a FLS to replace the term $\boldsymbol{K}sign(\boldsymbol{s}^*)$ in the equation (20). By this way, the chattering will be eliminated and the tracking error will be reduced also. The proposed adaptive synchronized sliding mode controller is described by the following equation:

$$\boldsymbol{\tau}_a = \boldsymbol{\tau}_{eq} - \boldsymbol{\Lambda}\boldsymbol{s}^* + \boldsymbol{\tau}_{Fuzzy} \tag{21}$$

in which $\boldsymbol{\tau}_{Fuzzy}$ is the output of a FLS for online learning the lump uncertainty $\Delta\boldsymbol{\tau}_a$. The FLS has the output described in (11).

3.3 Stability Analysis

The Lyapunov function candidate is chosen as:

$$V = \frac{1}{2}\boldsymbol{s}^*\hat{\boldsymbol{M}}_a\boldsymbol{s}^{*T} + \frac{1}{2}\tilde{\boldsymbol{W}}^T\eta^{-1}\tilde{\boldsymbol{W}} \tag{22}$$

where $\tilde{\boldsymbol{W}} = \boldsymbol{W}^* - \boldsymbol{W}$ is the error between the optimal value and the estimated value of $\boldsymbol{W}$.

Obviously, V is a positive definite function. The derivative of V is computed as follows:

$$\dot{V} = \frac{1}{2}\left(\dot{\boldsymbol{s}}^{*T}\hat{\boldsymbol{M}}_a\boldsymbol{s}^* + \boldsymbol{s}^{*T}\dot{\hat{\boldsymbol{M}}}_a\boldsymbol{s}^* + \boldsymbol{s}^{*T}\hat{\boldsymbol{M}}_a\dot{\boldsymbol{s}}^*\right) - \tilde{\boldsymbol{W}}^T\eta^{-1}\dot{\boldsymbol{W}} \tag{23}$$

By substituting the properties of the dynamic model of robot manipulators described in section 2 into the equation (23) we obtain:

$$\begin{aligned}\dot{V} &= \boldsymbol{s}^{*T}\hat{\boldsymbol{M}}_a\dot{\boldsymbol{s}}^* + \boldsymbol{s}^{*T}\hat{\boldsymbol{C}}_a\boldsymbol{s}^* - \tilde{\boldsymbol{W}}^T\eta^{-1}\dot{\boldsymbol{W}} \\ &= \boldsymbol{s}^{*T}\left(\boldsymbol{\tau}_a - \hat{\boldsymbol{C}}_a\dot{\boldsymbol{\theta}}_a^r - \hat{\boldsymbol{G}}_a - \Delta\boldsymbol{\tau}_a - \hat{\boldsymbol{M}}_a\ddot{\boldsymbol{\theta}}_a^r\right) - \tilde{\boldsymbol{W}}^T\eta^{-1}\dot{\boldsymbol{W}}\end{aligned} \tag{24}$$

Now, substituting the proposed controller (21) into the equation (24) we have:

$$\dot{V} = \boldsymbol{s}^{*T}\left(\boldsymbol{\tau}_{Fuzzy} - \Delta\boldsymbol{\tau}_a - \boldsymbol{\Lambda}\boldsymbol{s}^*\right) - \tilde{\boldsymbol{W}}^T\eta^{-1}\dot{\boldsymbol{W}} \tag{25}$$

There exist an ideal matrix $\boldsymbol{W}^*$ of the FLS such that:

$$\Delta\boldsymbol{\tau}_a = \boldsymbol{W}^{*T}\boldsymbol{\Phi}(\boldsymbol{x}) + \ni \tag{26}$$

where $\ni$ is bounded error vector.

From (11), (25) and (26) we have:

$$\dot{V} = \boldsymbol{s}^{*T}\left(\boldsymbol{W}^T\boldsymbol{\Phi}(\boldsymbol{x}) - \boldsymbol{W}^{*T}\boldsymbol{\Phi}(\boldsymbol{x}) - \ni - \boldsymbol{\Lambda}\boldsymbol{s}^*\right) - \tilde{\boldsymbol{W}}^T\eta^{-1}\dot{\boldsymbol{W}} \tag{27}$$

From (27), if we choose the online tuning law of the FLS:

$$\dot{\boldsymbol{W}} = -\eta\boldsymbol{\Phi}(\boldsymbol{x})\boldsymbol{s}^* \tag{28}$$

then the equation (27) could be rewritten:

$$\dot{V} = \boldsymbol{s}^{*T}\left(-\ni - \boldsymbol{\Lambda}\boldsymbol{s}^*\right) = \sum_{i=1}^{n}\left(-\ni_i - a_i s_i^*\right) \tag{29}$$

We assume that:

$$\left|\ni_i\right| \le \delta_i\left|s_i^*\right| \tag{30}$$

in which δ_i (i = 1,..,n) are positive constants which always can be found.

By substituting (30) into (29) we obtain:

$$\dot{V} \le \sum_{i=1}^{n} s_i^{*2}\left(-\delta_i - a_i\right) \tag{31}$$

Therefore, if we choose $a_i \le \delta_i$ then $\dot{V} \le 0$. It could be concluded that the overall system is asymptotically stable based on Lyapunov theory.

4. SIMULATION

To illustrate the proposed controller, the simulations are conducted for a 3 degrees-of-freedom (DOF) planar parallel manipulator working on a horizonal plane as depicted in Figure 2. The dynamic model of the parallel manipulator was described in [12].

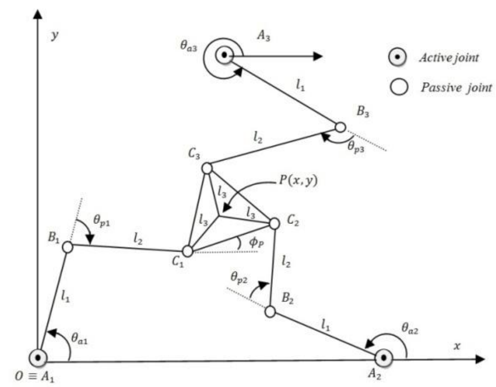

Figure 2. The 3 DOF planar parallel manipulator

Simulation studies were conducted on Matlab-Simulink and the mechanical model of the 3 DOF planar parallel manipulator was built on SimMechanics toolbox. Parameters in the mechanical model were set to be: l_1 = 0.4m, l_2 = 0.6m, l_3 = 0.2m, m_1 = 1kg, m_2 = 1.2kg, I_{z11} = I_{z21} =0.0033kgm^2, I_{z12} = I_{z22} = 0.0072kgm^2, r_1 = 0,2m, r_2 = 0.3m.

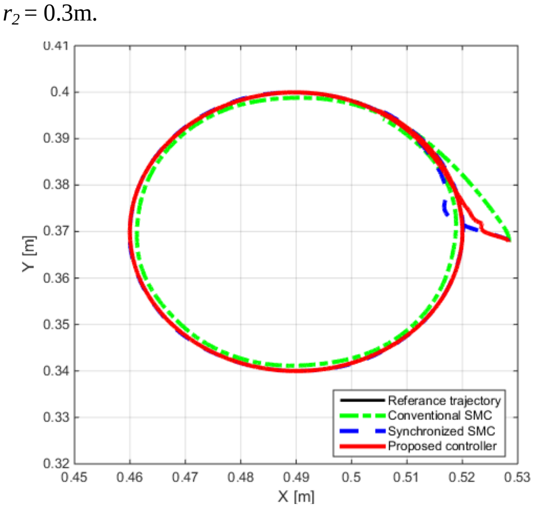

Figure 3. The results of tracking trajectory

The end-effector of the parallel manipulator is driven to track a circular trajectory on XY plane during 7 seconds. The proposed adaptive synchronized sliding mode controller is simulated and compared with two other controllers: Traditional sliding mode controller (SMC), the synchronized SMC and the proposed controller.

Figure 3 shows the results of tracking circular trajectory. It can be seen that, the proposed controller bring about the most quickly and exactly convergence (the red solid line) to the desire trajectory among three control methods.

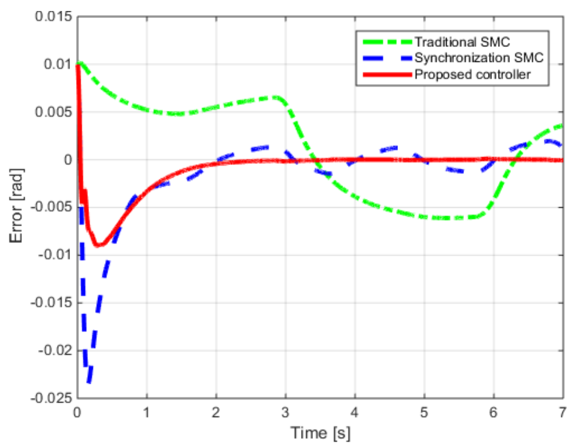

Figure 4. The tracking error of active joint *1*th

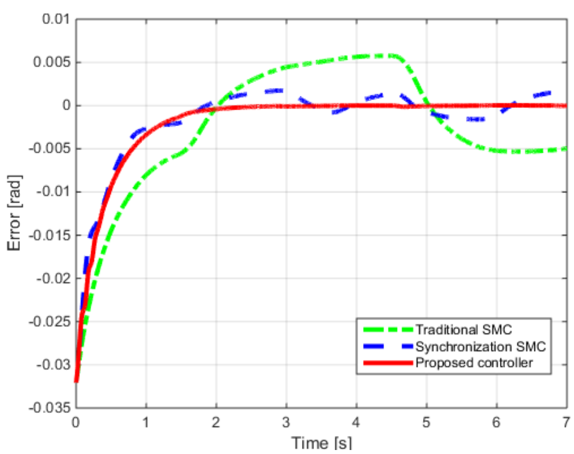

Figure 5. The tracking error of active joint *2*th

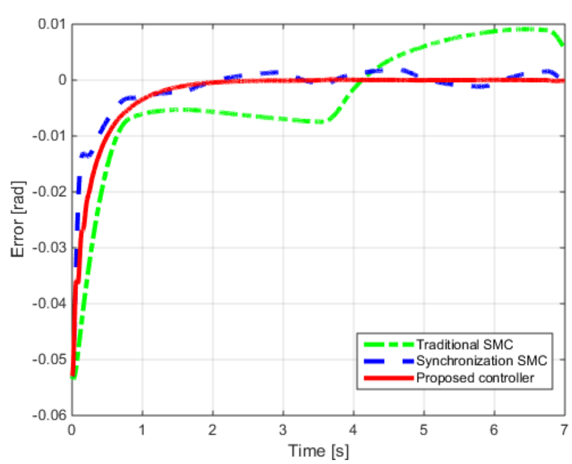

Figure 5. The tracking error of active joint *3*th

The tracking errors of active joints *1*th, *2*th and *3*th are shown in Figures 3, 4 and 5. We can easy recognize that the tracking errors caused by the proposed controller are smaller than that of the cases of using traditional SMC and synchronization SMC.

The tracking errors in X-direction and Y-direction are also displayed in Figures 6 and 7. The results showed that the proposed controller brings about the smallest tracking errors among three control methods. It can be concluded that the proposed controller have the most effectively performance in comparison to the traditional SMC method and synchronization SMC method.

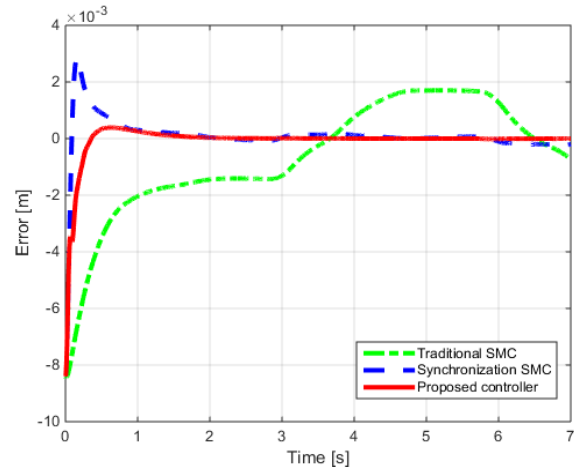

Figure 6. Tracking error X-direction

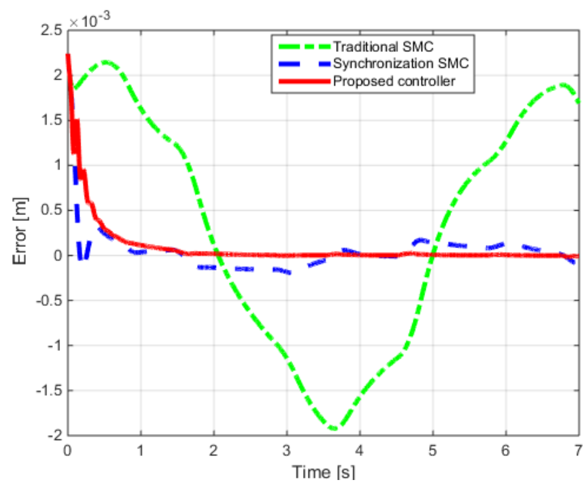

Figure 7. Tracking error Y-direction

5. CONCLUSION

In this paper, we presented the design of adaptive synchronized sliding mode controller for planar parallel manipulators. The dynamic model of parallel manipulators is presented in active joint space. Based on this dynamic model, and combine with the definitions of synchronized control, sliding mode control and online self-tuning fuzzy logic system, a synchronized sliding mode controller is developed. The self-tuning law is proposed by using the cross-coupling error of the parallel manipulator. The stability of the closed loop system is guaranteed by using Lyapunov theory. The simulations on Matlab/Simulink showed that the proposed control algorithm bring about the smallest errors in comparison to the traditional SMC and synchronization SMC without uncertainties compensation.

6. ACKNOWLEDGMENTS

This research is funded by the Ministry of Education and Training (MOET) of the Socialist Republic of Vietnam, under grant number KYTH-17.

7. REFERENCES

[1] Merlet, Jean-Pierre, and Clément Gosselin. "Parallel mechanisms and robots." *Springer Handbook of Robotics*. Springer Berlin Heidelberg, 2008. 269-285.

[2] Wu, Jun, et al. "Dynamics and control of a planar 3-DOF parallel manipulator with actuation redundancy." *Mechanism and Machine Theory* 44.4 (2009): 835-849.

[3] Su, Y. X., B. Y. Duan, and C. H. Zheng. "Nonlinear PID control of a six-DOF parallel manipulator." *IEE Proceedings-Control Theory and Applications* 151.1 (2004): 95-102.

[4] Kuo, Yong-Lin, Tsu-Pin Lin, and Chun Yu Wu. "Experimental and numerical study on the semi-closed loop control of a planar parallel robot manipulator." *Mathematical Problems in Engineering* 2014 (2014).

[5] Le, Tien Dung, Hee-Jun Kang, and Young-Soo Suh. "Chattering-free neuro-sliding mode control of 2-DOF planar parallel manipulators." *International Journal of Advanced Robotic Systems* 10.1 (2013): 22.

[6] Zhao, Dongya, Shaoyuan Li, and Feng Gao. "Fully adaptive feedforward feedback synchronized tracking control for Stewart Platform systems." *International Journal of Control, Automation, and Systems* 6.5 (2008): 689-701.

[7] Koren, Yoram. "Cross-coupled biaxial computer control for manufacturing systems." *Journal of Dynamic Systems, Measurement, and Control* 102.4 (1980): 265-272.

[8] Su, Yuxin, et al. "Integration of saturated PI synchronous control and PD feedback for control of parallel manipulators." *IEEE Transactions on Robotics* 22.1 (2006): 202-207.

[9] Sun, Dong, et al. "Synchronous tracking control of parallel manipulators using cross-coupling approach." *The International Journal of Robotics Research* 25.11 (2006): 1137-1147.

[10] Ren, Lu, James K. Mills, and Dong Sun. "Adaptive synchronized control for a planar parallel manipulator: theory and experiments." *Journal of dynamic systems, measurement, and control* 128.4 (2006): 976-979.

[11] Zhao, Dongya, Shaoyuan Li, and Feng Gao. "Fully adaptive feedforward feedback synchronized tracking control for Stewart Platform systems." *International Journal of Control, Automation, and Systems* 6.5 (2008): 689-701.

[12] Le, Quang Dan, Hee-Jun Kang, and Tien Dung Le. "An Adaptive Position Synchronization Controller Using Orthogonal Neural Network for 3-DOF Planar Parallel Manipulators." *International Conference on Intelligent Computing*. Springer, Cham, 2017.

[13] Zhao, Dongya, Shaoyuan Li, and Feng Gao. "Finite time position synchronised control for parallel manipulators using fast terminal sliding mode." *International Journal of Systems Science* 40.8 (2009): 829-843.

[14] Cheng, Hui, Yiu-Kuen Yiu, and Zexiang Li. "Dynamics and control of redundantly actuated parallel manipulators." *IEEE/ASME Transactions on mechatronics* 8.4 (2003): 483-491. DOI: 10.1109/TMECH.2003.820006

[15] Le, Tien Dung, and Hee-Jun Kang. "An adaptive tracking controller for parallel robotic manipulators based on fully tuned radial basic function networks." *Neurocomputing* 137 (2014): 12-23. DOI: 10.1016/j.neucom.2013.04.056

[16] Yoo, Byung Kook, and Woon Chul Ham. "Adaptive control of robot manipulator using fuzzy compensator." *Fuzzy Systems, IEEE Transactions on* 8.2 (2000): 186-199. DOI: 10.1109/91.842152

[17] Zhao, Dongya, et al. "Synchronized control of mechanical systems: a tutorial." *Applied Methods and Techniques for Mechatronic Systems*. Springer Berlin Heidelberg, 2014. 1-25.

[18] Zhao, Dongya, Shaoyuan Li, and Feng Gao. "Fully adaptive feedforward feedback synchronized tracking control for Stewart Platform systems." *International Journal of Control, Automation, and Systems* 6.5 (2008): 689-701.