Abstract
Different coating technological requirements of automobile electro-coating control system have different desired trajectories. For a dual parallel robot used for automobile electro-coating conveying (DPRAEC), an intelligent tracking smooth sliding mode synchronization control method (ITSSMSC) is proposed to achieve better tracking control performance with high synchronization and flexibility for the system under different desired trajectories. According to the dual parallel and symmetrical structure of the robot, the sliding surface is designed based on a composite error. A deep neural network with multi-hidden layer is trained by deep learning algorithms. Then, any given trajectories caused by different coating technological requirements could be tracked without tuning additional parameters and the synchronous performance could be guaranteed. The convergence of the network training process and the stability of the ITSSMSC are proved theoretically. Compared with the synchronization proportion-derivative control, the sliding mode control based on constant velocity reaching law without synchronization control and the smooth sliding mode synchronization control without training, simulation and experimental results demonstrate the effectiveness of the proposed approach. Compared with synchronization proportion-derivative control, the maximum tracking errors of constant velocity reaching law without synchronization control among each active joint are, respectively, reduced 92.2%, 94.7%, 98.8%. The maximum tracking errors of smooth sliding mode synchronization control and ITSSMSC in $z$ direction are, respectively, $9.06\times10^{-3}$m and $1.01\times10^{-3}$m. The maximum tracking errors of smooth sliding mode synchronization control and ITSSMSC in $\beta$ angle are, respectively, $4.50\times10^{-2}$rad and $1.45\times10^{-2}$rad. Therefore, the proposed control method could improve the tracking control performance and synchronization performance simultaneously under different desired trajectories.

Keywords
Parallel robot, intelligent control, synchronization control, sliding mode control, automobile electro-coating conveying

Introduction
The main existing automobile electro-coating conveying equipments, such as suspended conveyor, swing-rod conveyor, VarioShuttle, and Rodip, et al., exhibit the disadvantage of weak carrying capacity due to the serial structure, which leads to low flexibility of multi-vehicle. To this end, a dual parallel robot used for automobile electro-coating conveying (DPRAEC) was developed to further improve the load-carrying capacity and flexibility of the robot.

Recently, some difficulties still exist in the research of high performance tracking control of the DPRAEC, that is, 1) To improve the synchronization and robustness performance of the uncertain system simultaneously; 2) To meet the flexibility requirements under the demand of different coating processes without constantly adjusting the controller parameters or switching the hardware and software of the control system. This is caused by the following points. Firstly, the DPRAEC consists of multiple active joints with strong couplings. The asynchrony of branches will descend the tracking accuracy of the end-effector, and even result in serious
damage to the robot. In addition, the DPRAEC is a bilaterally symmetrical robot with parallel branches in each side, so the operation of robot is equivalent to the synchronous operation of two sets of parallel robots. Therefore, compared with general robots, DPRAEC has higher requirements for its synchronous coordination performance. In addition, due to the problem of synchronous coordination, the tracking control performance of DPRAEC may deteriorate due to the system uncertainties such as external disturbance and internal nonlinear friction. Secondly, on the aspect of actual production process, the desired trajectories of the DPRAEC should be defined according to the requirements of coating technological, that is, the desired trajectories of the DPRAEC will be different due to the different requirements of coating technological. The controller parameters need to be adjusted continuously to meet the ideal control requirements for different desired trajectories. While the flexible advantage of its mechanical mechanism for the researched parallel robot could not be exerted under the control of DPRAEC. Sliding mode control (SMC) has been adopted in practical engineering due to its advantages of being insensitive to external disturbance and internal uncertainty. However, when it comes to the dual parallel robot, due to the strong coupling among the joints, the synchronization should be considered as well. The tracking error-based SMC could not guarantee the trajectory tracking performance and the synchronization performance simultaneously. Existing researches combine synchronous control and SMC to enhance the synchronization performance of the system. A second-order SMC based on the deviation coupling synchronous control is proposed to improve the performance of synchronization control and ensure the fast convergence of the speed error, but the complexity of the control structure will grow as the motors in robot system increase. In Zhong et al., a global SMC strategy of adjacent cross-coupled synchronous control is developed to realize the position synchronous control of servo system, so as to improve the tracking and synchronization performance of the system. However, the synchronization error in Zhong et al. only takes into account the synchronization among adjacent axes. For DPRAEC, if the synchronization error in Zhong et al. is adopted, it is difficult to ensure better synchronous coordination performance of the robot when the branches on both sides of the robot are not synchronized or the motion transfer errors exist from the active joint to the end-effector. Therefore, a novel synchronization error that takes into account the synchronization of end-effector and both sides of the robot is defined for DPRAEC in this paper.

To track different trajectories without constantly adjusting the controller parameters, recently, deep learning has been applied in the field of trajectory tracking control. In order to make a multirotor UAV (unmanned aerial vehicle) learn the expert version of trajectory tracking, a reward function of minimum trajectory tracking error in data set is studied by inverse reinforcement learning, and a reward function-based reinforcement learning controller is designed. To handle the problems existing in multi-object tracking, and improve the tracking accuracy of multi-object, a robust trajectory tracking method based on Bayesian filtering within deep learning is proposed. The introduction of deep learning could achieve a well-trained deep neural network architecture without adjusting the controller parameters. This ensures that when the DPRAEC system needs to meet the requirements of different desired trajectories, the controller parameters do not need to be re-adjusted, so as to improve the robustness of the control system facing the changed desired trajectories.

Inspired by the research in synchronous SMC and deep learning, an intelligent tracking smooth sliding mode synchronization control method (ITSSMSC) by incorporating deep learning is proposed in this paper. The main contributions of this paper can be stated as follows.

1) The robustness of the system is enhanced by the chattering-free SMC. The synchronous coordination of the system is improved through the defined synchronization error between end-effectors of the dual parallel robot.
2) Any given trajectories caused by different coating technological requirements could be tracked without constantly adjusting the controller parameters, which will save debugging time.
3) The stability of the dual parallel robot control system is proved theoretically, which is not mentioned in other related papers.

This paper is organized as follows. The modeling process of the DPRAEC is introduced in Section II. Section III describes the controller design of the DPRAEC. In this section, the stability of learning process and the reliability of the system are also considered. The simulation results of the proposed controller are presented in Section IV together with the description of the experimental system and the experimental results of the proposed controller. Finally, the conclusions are summarized in Section V.

**DPRAEC and dynamic modeling**

**Physical description**

The DPRAEC is a bilaterally symmetrical mechanism mainly includes two function parts: a lifting-turning mechanism and a walking mechanism. The structure of the lifting-turning mechanism (LFM) is composed of two symmetrical hybrid mechanisms. Figure 1 presents the unilateral structure of DPRAEC. The parts labeled 1–9 and 12–13 in Figure 1 also exist on the other side of the robot, and the bilaterally mechanisms are connected symmetrically by a connecting rod 10. Figure 1 presents the unilateral structure of DPRAEC. As one of the
portions of DPRAEC, the control of walking mechanism is relatively simple without complex dynamic control. Therefore, this paper will focus on dynamic modeling and dynamic control for the LFM, the other main portions of DPRAEC.

Dynamic model

In consideration of the element in real operating environment, such as external disturbance and internal non-linear friction, the mathematical dynamic of the LFM can be written as:

\[ M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + J^T D(t) + J^T F(t) = Q \]  

where \( M(q) = \dot{M}(q) + \Delta M(q) \) is the inertial matrix, \( C(q, \dot{q}) = \dot{C}(q, \dot{q}) + \Delta C(q, \dot{q}) \) is the Coriolis and centrifugal force term, \( G(q) = \dot{G}(q) + \Delta G(q) \) is the gravity term, \( Q \) is the generalized driving force/torque, \( \Delta M(q), \Delta C(q, \dot{q}), \Delta G(q) \) represent the modeling errors existing in \( M(q), C(q, \dot{q}), G(q) \) respectively. \( q, \dot{q}, \ddot{q} \) represent the actual pose, actual velocity, and actual acceleration of the end-effector, respectively. \( q \) is defined as \( q = (z, \beta)^T \), with \( z \) and \( \beta \) are the position component of the end-effector, \( D(t) = F_s \text{sgn}(\dot{x}) + B_s \dot{x} \) and \( F(t) \) represent the friction and the external disturbance term, respectively. \( J \) represents the Jacobi matrix of the LFM.

For the robot dynamics researched in this paper, the above \( \dot{M}(q), \dot{C}(q, \dot{q}), \dot{G}(q) \) can be obtained by the following formulas: \( \dot{M}(q) \) is derived from

\[ T = \frac{1}{2} \dot{q}(t)^T \dot{M}(q) \dot{q}(t), \quad \dot{C}(q, \dot{q}) = \dot{M}(q) - \frac{1}{2} \dot{q}(t)^T \dot{M}(q), \quad \dot{G}(q) = \frac{\partial P}{\partial q}, \]

where \( T \) and \( P \) are the kinetic energy and potential energy of the system, respectively. In addition, \( T = T_1 + T_2 + T_3 + T_4 \), where \( T_i (i = 1, 2, 3, 4) \) represent the vehicle kinetic energy, branch chain kinetic energy, transmission kinetic energy and sliders potential energy, respectively. \( P = P_1 + P_2 + P_3 + P_4 \), where \( P_i (i = 1, 2, 3, 4) \) represent the vehicle potential energy, branch chain potential energy, transmission potential energy and sliders potential energy, respectively. Due to the space limitations, the specific contents of the above \( T_i (i = 1, 2, 3, 4) \) and \( P_i (i = 1, 2, 3, 4) \) can be found in Cao et al.\(^{12}\)

Remark 1. The researched dual parallel robot is a bilaterally symmetrical mechanism. Equation (1) is the mathematical model of the main part LFM for the above-mentioned dual parallel robot as a whole. In addition, due to the space limitations, one example is provided to demonstrate the bilaterally symmetrical structure characteristics of the established model, such as \( M_{12} \) is equal to \( M_{21} \) in the inertial matrix \( M(q) \) matrix component of the model.

For the purpose of realizing the actual control of the DPRAEC, the dynamic of DPRAEC in the task space needs to be converted to the joint space.

According to the relationship between the task space and the joint space,\(^{13}\) (1) can be rewritten as follows:

\[ M(x)\ddot{x} + C(x, \dot{x})\dot{x} + G(x) + D(t) + F(t) = \tau \]  

Obviously, for the above dynamic model, considering the characteristics of bilateral symmetric structure and the existence of external disturbance and internal friction, it can hardly meet the control requirements of high tracking accuracy and strong robustness via the conventional PID method.

Problem formulation

To consider the tracking performance of DPRAEC, the desired pose of the end-effector \( q_{\text{des}}(i), (i = 1, 2) \) should approach the actual pose \( q_i(t), (i = 1, 2) \), that is, \( \forall n \lim_{n \to \infty} (q_{\text{des}}(i) - q_i(t)) = 0, (i = 1, 2) \).

Then, the pose tracking error in two sides of the end-effector is defined as \( e_i(t) = q_{\text{des}}(i) - q_i(t), (i = 1, 2) \), the synchronization error is given by:

\[ e(t) = e_1(t) - e_2(t) \]  

Remark 2. If \( e(t) \to 0 \), the two sides of the end-effector are synchronous.

Additionally, the controller parameters need to be adjusted according to the different desired trajectories caused by the different coating technological requirements. In order to realize the intelligent trajectory tracking control of the above control requirements, an ITSSMSC is designed. The main design process is as follows: By constructing a deep neural network with multiple hidden layers, the actual motion trajectories are...
taken as the inputs under the control of SSMSC with different expected motion trajectories. And the torque outputs of SSMSC are used as the output samples. The deep learning algorithm is introduced and the ReLU function is selected as the activation function to learn and train the network. It is expected that any given desired motion trajectories are used as inputs. Then, the required torque output of ITSSMSC could be obtained to achieve high-performance trajectory tracking control without additional parameter adjustment.

The purpose of this paper is to design an intelligent tracking smooth sliding mode synchronization controller, so as to achieve the following goals: 1) The actual pose \( q(t) \) can track any desired pose \( q_d(t) \) caused by different coating technological requirements without additional controller parameters tuning; 2) the synchronous error (4) meet the condition given by Remark 2.

Controller design

As mentioned in section II, there exists disturbance in the actual environment of DPRAEC, these factors will more or less affect the control performance of the robot. Under such circumstances, a comprehensive control scheme to overcome the adverse effects of the above factors is designed. Figure 2 shows the control block diagram, where Part A represents the ITSSMSC method.

In Figure 2, the desired pose \( \dot{x}_d \) and velocity \( \ddot{x}_d \) of each active joint are obtained by calculating inverse kinematics of the desired trajectory of DPRAEC’s end-effector. The tracking error \( e \) of the end-effector and the tracking error \( \dot{e} \) of the active joints are obtained by calculating with the actual pose and velocity \( \dot{x} \) of each active joint. The deep neural network takes the desired trajectory \( x_d, \dot{x}_d \) as input, and trajectory tracking error as the weight adjustment rule. The output of the deep neural network is the control torque of each active joint. The dynamic control of DPRAEC is realized under the control of the proposed control method.

SSMSC

According to the kinematics analysis, a synchronization error is given by:

\[
e(t) = He_x(t)
\]

where \( e_x(t) \) is the pose tracking error of each joint, \( H \) is related to the transformation relationship between the active joints and the end-effector, \( e_{\phi}(t) \) \((i = 1, 2, 3, 4)\), \( e_{\phi}(t) \) \((j = 1, 2)\) and \( H \) can be respectively given by:

\[
e_{\phi}(t) = \dot{x}_d(t) - x(t)
\]

\[
e_{\phi}(t) = \dot{x}_d(t) - x_d(t)
\]

\[
H = R_1V_1 - R_2V_2
\]

where \( x_d(t) \) and \( \phi_d(t) \) represent the desired position and angle of each active joint, respectively. \( x_d(t) \) and \( \phi_d(t) \) represent the actual position and angle of each active joint, respectively. The matrices of \( R_i \) \((i = 1, 2)\) and \( V_i \) \((i = 1, 2)\) are described by:

\[
R_1 = \begin{bmatrix}
\frac{x_2 - x_1}{4z_1} & \frac{x_3 - x_1}{4z_2} & 0
\
0 & 0 & \frac{z_1}{n}
\end{bmatrix}
\]

\[
R_2 = \begin{bmatrix}
\frac{x_2 - x_1}{4z_1} & \frac{x_3 - x_1}{4z_2} & 0
\
0 & 0 & \frac{z_1}{n}
\end{bmatrix}
\]

\[
V_1 = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0
0 & 1 & 0 & 0 & 0 & 0
0 & 0 & 0 & 0 & 1 & 0
\end{bmatrix}
\]

\[
V_2 = \begin{bmatrix}
0 & 0 & 1 & 0 & 0 & 0
0 & 0 & 0 & 1 & 0 & 0
0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
\]

Figure 2. Control block of DPRAEC.
Then, the composite error is defined in combination (3) and (4) as follows:

$$
\epsilon^*(t) = \epsilon(t) + \lambda \int_0^t \epsilon(t) dt \quad (11)
$$

where \( \lambda = \text{diag}(\lambda_1, \lambda_2) \) is the coupling matrix between each joint.

From the designed composite error (12), we design a sliding surface function as follows:

$$
S = \epsilon^*(t) + A\epsilon^*(t) \quad (12)
$$

where \( A = \text{diag}(a_1, a_2) \), \( a_1, a_2 \) are two designed constants and meet the Routh-Hurwitz stability criterion.

Based on the composite error in (11), we can rewritten (12) as the following form:

$$
\dot{S} = \dot{\epsilon}^*(t) + A\dot{\epsilon}^*(t) = A\epsilon(t) + \lambda A\epsilon(t) + \ddot{q}_d(t) - \ddot{q}(t) + \lambda \ddot{z}
$$

In order to diminish the chattering caused by the non-continuous sign function \( \text{sgn}(S) \) in the constant velocity reaching law, the following continuous function given below is used to replace the non-continuous function:

$$
f(S) = \frac{S}{\| S \|} \quad (13)
$$

where, \( S = [s_1, s_2]^T \), \( \| S \| = (s_1^2 + s_2^2)^{1/2} \).

**Theorem 1.** Considering the dynamic model of the LFM (1), with the sliding mode variable (12) and continuous function (14), if the SSMSC control law is given by:

$$
Q_{\text{SSMSC}} = M(q)(A\dot{e}(t) + \lambda A\dot{e}(t) + \ddot{q}_d(t) + K_3 S + \lambda \ddot{z}(t)) + C(q, \dot{q})\ddot{q} + G(q) + J^T D(t) + J^T F(t) \quad (15)
$$

where \( K = \text{diag}(k_1, k_2) \), \( k_1 > 0, k_2 > 0 \).

The system (1) will be asymptotically stable.

**Proof.** Consider a Lyapunov function as:

$$
V_3 = \frac{1}{2} S^T S \quad (16)
$$

Differentiating (16) with respect to time under the control law in (15), we have:

$$
\dot{V}_3 = S^T \dot{S} = S^T [A\dot{e}(t) + \lambda A\dot{e}(t) + \ddot{q}_d(t) - \ddot{q}(t) + \lambda \ddot{z}]
$$

$$
= S^T [A\dot{e}(t) + \lambda A\dot{e}(t) + \ddot{q}_d(t) - M(q)^{-1}(Q - C(q, \dot{q})\ddot{q} - G(q) - J^T D(t) - J^T F(t) + \lambda \ddot{z}(t)]
$$

$$
= -S^T K \frac{S}{\| S \|}
$$

$$
= -K \| S \|
$$

(17)

where \( K \) is a positive-defined matrix and \( \| S \| \geq 0 \), which make \( \dot{V}_3 \) satisfy the condition \( \dot{V}_3 \leq 0 \). Thus, we conclude that the system (1) will be asymptotically stable along SSMSC (15). This completes the proof.

**ITSSMSC**

For the purpose of tracking any desired trajectories caused by different coating technological requirements intelligently, an ITSSMSC strategy by introducing the deep learning algorithm is proposed. The SSMSC is replaced by the trained deep neural network controller. Firstly, the feature extraction of large-capacity sample data of shallow neural network with single hidden layer is insufficient, which limited the modeling capability for complex and nonlinear objects. So a deep neural network with strong nonlinear and large-capacity trained by better learning the control input and output data is constructed. Secondly, in order to solve the phenomenon of vanishing gradients happening during the process of back propagation of conventional BP neural network, the ReLU activate function is adopted, the ReLU function is given as follows:

$$
f(Z) = \begin{cases} 
Z, Z \geq 0 \\
0, Z < 0 
\end{cases} \quad (18)
$$

Under the premise of the unsaturation of the control input, the control input and output data under different desired trajectories are sampled as the training datasets by changing the velocity of walking motion and the amplitude of lifting and turning motion of the parallel robot.

1) **Overall Architecture of the deep neural network:** The deep neural network designed in this paper is based on the deep learning algorithm. Specifically, the deep neural network is the network structure and the deep learning is the learning algorithm used to training the network structure. In order to reflect the advantages of deep learning algorithm in network depth and the ability to learn strong nonlinear large-capacity data by selecting the activation function, as well as to reflect the clear and


intuitive neural network architecture used in this paper. The structure of deep neural network is shown in Figure 3. In the figure, the number of units in input layer and output layer is \( i = 6, j = 6 \), respectively; the number of units in hidden layers is \( \nu^l = 22 \), the superscript \([l]\) represents the \( l\)th layer; \( L \) is the number of layers of neural network, and the number of hidden layers is \( L - 1 = 6 \). The input is matrix data: independent variable \( X \) is different desired trajectories of end-effector of DPRAE C, dependent variable \( Y \) is the corresponding force/torque of each active joint of SSMSC. Then, the data go through linear and ReLU layer, finally \( \hat{Y} \) output through linear and Sigmoid layer.

2) Training: The training takes place on PC with Intel core i5-8400 CPU. Script programming is used to train deep neural network in MATLAB. The initialization in He et al.\(^{16}\) and ReLU function are used to get better training results. The number of iterations in gradient descent loop is \( N = 2000 \), the constant learning rate is \( \eta = 0.12 \).

Some issues regarding computational cost of the proposed control method have been investigated in Remark 3.

**Remark 3.** For now, one of the deficiency of both the sigmoid function and tanh function is that if the output \( z \) of the linear part layer is either very large or very small, then the gradient of the derivative or the slope of the function will become very small, which can slow down gradient descent. However, the adoption of ReLU function in our manuscript can ameliorate the above problem. Since for a lot of the space of \( z \), the derivative or the slope of the ReLU function is very different from zero, which will make the neural network learn much faster than using the sigmoid or the tanh function, that is, the ReLU function has less effect on the slope of the function going to zero which slows down learning speed. Although when \( z \) is equal to zero, the derivative of the function is not well-defined, the odds of obtaining \( z \) exactly equal to zero is small in the actual programming process, and it can be assigned to 0 or 1 when \( z \) is equal to 0.

**Theorem 2.** If the above algorithms and parameters settings are used to implement the neural network training, then the convergence of the training process is ensured.

**Proof.** At first, we stipulate that \( W = [W^1, W^2, \ldots, W^L] \) and \( A = [A_1, A_2, \ldots, A^L] \) represent the weight matrix and the output value of each layer, respectively.

The Lyapunov function is given by:

\[
E(k) = \frac{1}{m} \| e(k) \|^2 = \frac{1}{m} \| Y(k) - \hat{Y}(k) \|^2
\]

where \( k \) represents the \( k\)th iteration.

Then, the following equation can be obtained from (19):

\[
\Delta E(k + 1) = E(k + 1) - E(k) = \frac{1}{m} \| e(k + 1) - e(k) \|^2
\]

where \( e(k + 1) \) can be expressed as

\[
e(k + 1) = e(k) + \frac{\partial e(k)}{\partial W} \Delta W
\]

where \( \Delta W = \eta W^l = \eta e(k) \frac{\partial \hat{Y}(k)}{\partial W} \).

By substituting (21) into (20), one knows:

\[
\Delta E(k + 1) = \frac{1}{m} \| e(k + 1) \|^2 \left( 1 - \eta \left[ \frac{\partial \hat{Y}(k)}{\partial W} \right]^T \left[ \frac{\partial \hat{Y}(k)}{\partial W} \right] \right)^2 - 1
\]

\[
= \frac{1}{m} \| e(k) \|^2 \left( 1 - \eta \left[ \frac{\partial \hat{Y}(k)}{\partial W} \right]^2 \right) - 1
\]

\[
= -\frac{1}{m} \| e(k) \|^2 \eta \alpha(k)
\]

where \( \alpha(k) = \eta \left\| \frac{\partial \hat{Y}(k)}{\partial W} \right\|^2 \left( 2 - \eta \left\| \frac{\partial \hat{Y}(k)}{\partial W} \right\|^2 \right) \).

**Output layer.** For the proposed neural network, the activate function of output layer is adopted as:

\[
f(x) = 1 / (1 + e^{-x})
\]

Considering the fact that \( 0 < \dot{f} \leq 1/4 \) and the definition of Euclidean norm, the following inequality can be obtained:

\[
0 < \left\| \frac{\partial \hat{Y}(k)}{\partial W} \right\| = \left\| \frac{\partial \hat{Y}(k)}{\partial Z^l} \right\| = \left\| f(x) \cdot A^l \right\| \leq \sqrt{\dot{f} \cdot (n^l - 1)^L \cdot \frac{1}{4} \max(A^l)}
\]
By substituting (24) into (22) and considering $\eta$ is usually chosen as a relatively small positive real number, we obtain:

$$\alpha(k) > 0$$

(25)

Thus, the following inequality can be obtained:

$$\Delta E(k + 1) < 0$$

(26)

**Hidden layer.** The proof process is similar to that of the output layer, according to (18), we can see that $f(Z) = 1 > 0$ if $Z > 0$, and thus $\alpha(k) > 0$ is ensured, then we can also conclude that $\Delta E(k + 1) < 0$, that is, $E(k + 1) < E(k)$.

Considering to the fact that $E(k)$ in (19) always be positive and according to (26), we can conclude from Lyapunov stability theory\(^7\) that when $k \to \infty$, we have $E(k) \to 0$, then the training error will satisfy the condition $e(k) \to 0$, that is, the convergence of the training process is ensured. This completes the proof.

Based on the proposed deep neural network, the control law of system (1) is designed as:

$$u = \hat{y} = \phi(x_d, W)$$

(27)

where $\|d\phi / dW \| \leq M, \exists M > 0$; the update law of weight is designed as:

$$\dot{W} = -\eta \left[ \frac{d\phi}{du} \frac{d\phi}{dW} \right]^T E_t$$

(28)

where $E_t = x(t) - x_d(t)$; for $\forall u$, there exists $x = h(u)$ and $\|d\phi / du\| \leq b, b > 0$, $h(\cdot)$ is a continuous function.

From the properties of BP neural network, there exists the desired weight $W^*$, one knows:

$$Q_{SSMSC} = \phi(x_d, W^*) + \eta_0$$

(29)

where $\eta_0$ is a minimum approximation error, and $\|\eta_0\| \leq \sup_{x_d, x} \|Q_{SSMSC} - \phi(x_d, W^*)\|$.

By substituting (27) into (1), add and subtract (16) on the right side of the substituted equation, one knows:

$$M \dot{e} + M \left( A \dot{e} + \lambda \dot{e} \right) + K \frac{S}{\|S\|} + \lambda \dot{e} = Q_{SSMSC} - \dot{\hat{y}}$$

(30)

According to (13), (30) can be rewritten as follow:

$$M \left( \dot{\hat{y}} + \frac{S}{\|S\|} \right) = Q_{SSMSC} - \dot{\hat{y}}$$

(31)

By substituting (27) and (29) into (31), the dynamic error formula of system (1) can be expressed as (32):

$$\dot{M} \hat{y} = -\phi(x_d, \xi) - MK \frac{S}{\|S\|} + \eta_0$$

(32)

where $\phi(x_d, \xi) = \phi(x_d, W) - \phi(x_d, W^*)$ and $\xi = [W - W^*]$. 

**Theorem 3.** Considering the dynamic error formula (32), if $\eta_0 = 0$ and the control law (27) with the update law of weight shown in (28) is adopted, then for any bounded initial conditions, it can be guaranteed that $S \to 0$ and $W \to W^*$.

**Proof.** The Lyapunov function is defined as follows:

$$V_4 = \frac{1}{2} S^T M S + \frac{1}{2} \xi^T \xi$$

(33)

The $V_4$ is divided into two parts for analysis:

1) The first part is defined as $\dot{z}_1 = \frac{1}{2} S^T M S$, differentiating $\dot{z}_1$ with respect to time and combining (32), one knows:

$$\dot{z}_1 = S^T M \dot{S}$$

$$= S^T \left[ -\phi(x_d, \xi) - MK \frac{S}{\|S\|} + \eta_0 \right]$$

$$\leq -M K \left( \xi^2 + s^2 \right) + S^T \eta_0$$

(34)

where $K$ and $M$ are both positive-defined matrix, when $\eta_0 = 0$, then $\dot{z}_1 \leq 0$.

2) The second part is defined as $\dot{z}_2 = \frac{1}{2} \xi^T \xi$, differentiating $\dot{z}_2$ with respect to time using the update law in (28), one knows:

$$\dot{z}_2 = -\eta \xi^T \left[ \frac{d\phi}{du} \right]^T \left[ h(u) - x_d \right] - \eta \xi^T \left[ \frac{dh(u)}{du} \frac{d\phi}{du} \right]^T \left[ h(u) - x_d \right]$$

(35)

According to the Lagrange mean value theorem, the expression $[h(u) - q_d]$ in (35) can be rewritten as follows:

$$h(u) - x_d = h(\phi(x_d, W)) - h(\phi(x_d, W^*))$$

$$= \frac{d\phi}{du} \frac{d\phi}{dW} \left( W - W^* \right)$$

(36)

where $W_1 = \mu W + (1 - \mu) W^*, 0 \leq \mu \leq 1$.

Then, (35) can be rewritten as:
\[ \dot{z}_2 = -\eta_2^T \frac{\partial h}{\partial \theta}^T \frac{x - h(u)}{w} - \eta_2^T \frac{\partial h}{\partial \theta}^T \frac{[h(u) - \dot{x}_d]}{w} - \eta_2^T \left( \frac{\partial h}{\partial \theta}^T \right) [h(u) - \dot{x}_d] \]

(37)

According to (27) and (28), the equation (37) satisfies the following inequality:

\[ \dot{z}_2 \leq - (\eta \|h(u) - \dot{x}_d\|^2 + 2\eta h M \|W - W^*\| \cdot \|h(u) - \dot{x}_d\| + \eta h B M \|W - W^*\| \cdot \|x - h(u)\|) < 0 \]

(38)

From the above two parts analysis, we have \( \dot{V}_4 = \dot{z}_1 + \dot{z}_2 \leq 0 \), it can be concluded that the system (1) has Lyapunov stability. This completes the proof.

### Simulation results

This section is scheduled as follows, the superiority of SSMSC control performance is verified first, and then the trajectory tracking performance of ITSSMSC and SSMSC under any desired trajectories caused by different coating technological requirements without parameter tuning are compared by simulations. The simulations are carried out in MATLAB 2017b environment with ode5 solver with a step size of 0.01. Among them, ode5, as a solver suitable for most continuous or discrete systems, can meet the solution requirements of the model simulation of the parallel robot system researched in this paper. At the same time, the simulation step size is appropriately reduced to 0.01 to meet the simulation accuracy requirements.

The synchronization proportion-derivative control (SPD) and the sliding mode control based on constant velocity reaching law (CVRLSMC) are also implemented for the comparison, the SPD control is designed as:

\[ Q_{\text{spd}} = M(q) (\dot{q}_d + K_d \dot{e}(t)) + K_p e(t) + C(q, \dot{q}) \dot{q} + G(q) + J^T D(t) + J^T F(t) \]

(39)

where \( K_p = \text{diag}(20, 20), K_d = \text{diag}(4, 4) \).

The CVRLSMC control is designed as:

\[ Q_{\text{CVRLSMC}} = M(q) (A e(t) + \lambda A \dot{a}(t) + \dot{q}_d(t)) + K_s g(S(t)) + K_p e(t) + C(q, \dot{q}) \dot{q} + G(q) + J^T D(t) + J^T F(t) \]

(40)

The uncertainty of parameters in (15), (39) and (40) are selected as \( F_c = \text{diag}(3.5, 3.5, 3.5, 3.5, 3.5, 3.5, 3.5, 3.5, 3.5, 3.5), B_c = \text{diag}(0.6, 0.6, 0.6, 0.6, 0.6, 0.6, 0.6, 0.6, 0.6, 0.6), F(t) = (3 \sin (2\pi t), \sin (2\pi t)) + 5 \text{randn} (2, 1) \), and the controller parameters in (15) and (40) are chosen as \( a_1 = 9, a_2 = 18, k_1 = 8, k_2 = 8, A_1 = 8, A_2 = 8 \), the desired trajectory is set as follows:

\[
\begin{align*}
z &= \begin{cases} 
0.25 & (0s \leq t \leq 2s) \\
0.2 + 0.05 \cos[2\pi(t - 2)] & (2s \leq t \leq 4s) \\
0.25 & (4s \leq t \leq 6s)
\end{cases} \\
\beta &= \begin{cases} 
\pi & (0s \leq t \leq 2s) \\
\pi - \pi \cos(0.25\pi t) & (2s \leq t \leq 4s) \\
\pi - \pi \sin(0.25\pi t) & (4s \leq t \leq 6s)
\end{cases}
\end{align*}
\]

(41)

The velocity \( v \) of walking motion and the amplitude \( A \) of lifting and turning motion, that is, \( v \in [0 \text{m/s}, 0.5 \text{m/s}], A \in [0.1 \text{m}, 0.5 \text{m}] \), in the desired trajectory are determined by the technological standards of the automobile coating and the size of the prototype. Within the above range of the velocity of walking motion and the amplitude of lifting and turning motion, different desired trajectories are defined by combining different velocities and different amplitudes randomly. Then the steady-state response performance of the control algorithm for different desired trajectories is verified. Furthermore, when \( v \) and \( A \) of the desired trajectory are required to be maximum, which greatly affects the steady-state reaching time and tracking accuracy of the control system. Due to the space limitation, thus parameters of the typical desired trajectory are selected as \( v = 0.5 \text{m/s}, A = 0.5 \text{m} \), that is, \( v \) and \( A \) are selected as maximum. And the typical desired trajectory of the parallel robot is defined as (42), to verify the steady-state response performance of the proposed approach.

\[
\begin{align*}
\dot{z}_2 \leq - (\eta \|h(u) - \dot{x}_d\|^2 + 2\eta h M \|W - W^*\| \cdot \|h(u) - \dot{x}_d\| + \eta h B M \|W - W^*\| \cdot \|x - h(u)\|) < 0
\end{align*}
\]

(38)

Figures 4 to 7 show the simulation results including trajectory tracking, synchronization error curves, driving force/torque and trajectory tracking error curves. The following facts have been noted: (a) Examining Figure 4,
it can be seen that the SSMSC has better trajectory tracking performance. Besides, compared with SPD, we clearly see from Table 1 that the SSMSC has smaller maximum tracking error (MTE) and mean square error of each active joint (AJMSE) during the whole motion process, which indicates the robustness properties of the proposed method under uncertainties. (b) As can be seen from Figure 5, compared with CVRLSMC without synchronous control technology, the synchronization error can be greatly decreased by applying the SSMSC, which will effectively improve the synchronous coordination performance of the system, and the corresponding data analyses in Table 2 also prove the above facts. As shown in Figure 6, the problem of chattering is solved due to the use of continuous analogy of sign function in (14).

(c) Figure 7 shows the trajectory tracking error curves of SSMSC and ITSSMSC under desired trajectory in (42). Due to the DPRAEC is conducting complex lifting and turning motions between 2s and 4s, for example, sinusoidal motion and cosinusoidal motion, the tracking errors are relatively large than that of 0–2s and 4–6s under the walking motions. It is worth noting that the order of magnitude is $10^{-4}$ in Figure 4. Therefore, the maximum tracking error within 2–4s could still ensure the smooth operation of the control system in the actual environment. The corresponding data analyses are listed in Table 3. Examining Table 3, we can find that since the introduction of deep learning algorithm to learn the control input and output under different desired trajectories, the ITSSMSC enables the position components of end-effector in $z$ direction and $\beta$ angle to obtain higher trajectory tracking accuracy than SSMSC. Remark 4 shows the relationship between the designed parameters and the control performance.

**Remark 4.** It is worth noting that the proposed control parameters should be selected for practical implementations carefully, since the control performance is often affected by the control input saturation, unmodeled system dynamics and environmental noise. The corresponding selection criteria of the control parameters are: (1) Generally, large values of $K$ can improve the convergence performance of SMC, but too large ones will induce the control saturation and chattering. For practical application, we need to carefully investigate the possible uncertainties and their amplitude that the...
actual system may experience. (2) The impact of $e$ is increasing with the value of $l$, the larger value of $l$, the better performance of synchronization control can be attained. The parameter $l$ can be designed through the trial and error method. (3) When determining the values of $n = \frac{10^{-4}}{C_0}$ and $L$, we need to consider the impact of the amount of hidden layers on training process. If the amount of hidden layers is small, the hidden layers contain less training sample information, which leads to poor training effect. On the contrary, when the amount of hidden layers is large, it is easy to produce overfitting, which makes the network training process easy to fall into local extremum. In addition, the network will appear underfitting when the number of units in hidden layers is too small.

Figure 6. Driving force/torque of the corresponding motor of each active joint: (a) first slider, (b) second slider, and (c) first driving wheel.

Figure 7. Tracking error of end-effector: (a) tracking error in $z$ direction and (b) tracking error in $\beta$ angle.

Table 1. MTE and AJMSE.

| Method   | MTE first slider ($\times 10^{-4}$m) | Second slider ($\times 10^{-4}$m) | First driving wheel ($\times 10^{-4}$rad) | AJMSE |
|----------|-------------------------------------|----------------------------------|------------------------------------------|-------|
| SPD      | 40                                  | 59                               | 4013                                     | 0.23  |
| SSMSC    | 3.1102                              | 3.1029                           | 3.6485                                   | 3.12 $\times 10^{-4}$ |

Table 2. Maximum synchronization error (MSE) and mean square error of end-effector (MSEE).

| Method   | MSE $z$ direction ($\beta$ angle (m/rad)) | MSEE |
|----------|------------------------------------------|------|
| CVRLSMC  | $7.7632 \times 10^{-10} / 3.74 \times 10^{-11}$ | $5.35 \times 10^{-10}$ |
| SSMSC    | $1.0023 \times 10^{-15} / 1.08 \times 10^{-14}$ | $7.35 \times 10^{-15}$ |

Table 3. Maximum tracking error (MTE) and MSE.

| Method   | MTE $z$ direction ($\beta$ angle (m/rad)) | MSEE |
|----------|------------------------------------------|------|
| SSMSC    | $9.0603 \times 10^{-3} / 0.0045$         | $3.18 \times 10^{-3}$ |
| ITSSMSC  | $1.0064 \times 10^{-9} / 1.4511 \times 10^{-4}$ | $0.11 \times 10^{-3}$ |
Experimental verification

The experiment platform of DPRAEC is demonstrated in Figure 8. The experiment platform constructs the distributed computer control system in the form of “PC + UMAC,” and UMAC communicates with PC in real-time through the Ethernet interface. The PC is equipped with Intel core CPU, 8G memory, 1TB storage space. It is used to implement the functions of data processing, code-writing, etc. The UMAC is used to implement the motion control of the robot, obtain the instructions and encode information acquisition, etc. The position detection equipment adopts the LJ18A3-8-Z-BY third-line proximity switch. The main function of the experiment platform is to realize the control for DPRAEC by developing the corresponding motion control applications and using the powerful function of UMAC. During the real-time electro-coating experiment, the users send control commands to the UMAC through the upper-level application program in the PC shown in Figure 8. After the UMAC receives the commands, the servo controller drives the servo motor of the DPRAEC to complete the electro-coating task. The desired trajectory is given by (43). The ITSSMSC and SSMSC controller parameters setting are the same as Section IV. The experiment results are shown in Figure 9.

Analyzed from Figure 9, the experimental tracking errors of end-effector under the control of ITSSMSC are smaller than that of SSMSC. The principal reason for this phenomenon is the relatively strong adaptability of any given desired trajectories caused by different coating technological requirements. And the proposed ITSSMSC could track different desired trajectories without tuning additional controller parameters. In practical application, the ITSSMSC could make the operation of DPRAEC more smoothly, which is conducive to improving the quality of automobile electro-coating and extending the working life of the robot.

Remark 5 shows the main advantages of the results given in Section III.

**Remark 5.** As the above simulation and experimental results, the main advantages of the results for practical applications are as follows: 1) Since DPRAEC is a dual parallel robot with symmetrical sides, the synchronization coordination performance of the robot is important. The results proposed in Section III show the improvement of synchronization, which is beneficial to improve the trajectory tracking accuracy. Meanwhile, the seriously damage is avoided due to large synchronization errors, such as deformation and distortion of the robot. 2) The results proposed in Section III effectively reduce the amplitude of chattering. 3) The
parameters adjustment is a time-consuming process due to the selection of different controller parameters under different desired trajectories. The ITSSMSC is proposed in Section III, which can track different desired trajectories without constantly parameters adjustment. So it could effectively improve the efficiency of parameters debugging.

Conclusion

In this paper, an ITSSMSC scheme was proposed for DPRAEC, it completely included dynamic modeling, controller design, simulation analysis, and experimental verification. From the paper, SSMSC owns the superiorities of improving the synchronization coordination, enhancing the robustness of the system and reducing the chattering. Since the ITSSMSC is proposed based on SSMSC, it also has the same superiorities in theory. In addition, due to the learning of SSMSC input and output under multi-desired trajectories, the ITSSMSC can track any desired trajectories caused by different coating technological requirements better than SSMSC without tuning the controller parameters, that is, the flexibility of the system has been increased.

As a future work, the method of more efficient acquisition of larger training data capacity is worth studying, and the neural network structure and learning algorithm can be further optimized to meet the requirements of nonlinear relationships between input and output under large data sets.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported in part by the National Natural Science Foundation of China (No. 51375210), the Zhenjiang Municipal Key Research and Development Program (No. GIZ2018004), the Graduate Research and Innovation Projects of Jiangsu Province (No. KYCX18_2252), and the Priority Academic Program Development of Jiangsu Higher Education Institutions (No. PAPD-2018-87).

ORCID iDs

Guoqin Gao https://orcid.org/0000-0002-5514-7822
Junwen Zhong https://orcid.org/0000-0002-9299-9047

References

1. Chen X, Wu J, Wang L, et al. Optimum design of a parallel conveyor for electrocoating of vehicle bodies. Int J Robot Autom 2016; 31(3): 198–205.
2. Cao Y, Gao G and Wu X. Chattering-free sliding mode synchronization control for a hybrid mechanism. Adv Mech Eng 2017; 9(7): 1–11.
3. Saab SS and Ghanem P. A multivariable stochastic tracking controller for robot manipulators without joint velocities. IEEE Trans Automat Contr 2018; 63(8): 2481–2495.
4. Liu XJ, Xie F and Wang J. Current opportunities in the field of mechanisms in China. J Mech Eng 2015; 51(13): 2–12.
5. Zhang G, Wang Y, Wang J, et al. Disturbance observer-based super-twisting sliding mode control for formation tracking of multi-agent mobile robots. Meas Control 2020; 53(5–6): 908–921.
6. Niu H, Lan Q, Liu Y, et al. A continuous integral terminal sliding mode control approach for a class of uncertain nonlinear systems. Meas Control 2019; 52(5–6): 720–728.
7. Li L, Sun L and Zhang S. Mean deviation coupling synchronous control for multiple motors via second-order adaptive sliding mode control. ISA Trans 2016; 62: 222–235.
8. Zhong G, Shao Z, Deng H, et al. Precise position synchronous control for multi-axis servo systems. IEEE Trans Ind Electron 2017; 64(5): 3707–3717.
9. Choi S, Kim S and Jin Kim H. Inverse reinforcement learning control for trajectory tracking of a multirotor UAV. Int J Control Autom Syst 2017; 15(4): 1826–1834.
10. Xiang J, Zhang G and Hou J. Online multi-object tracking based on feature representation and Bayesian filtering within a deep learning architecture. IEEE Access 2019; 7: 27923–27935.
11. Deepa SN and Baranilingesan I. Optimized deep learning neural network predictive controller for continuous stirred tank reactor. Comput Electr Eng 2018; 71: 782–797.
12. Cao Y, Gao G and Wu X. Synchronization control based on novel error for a hybrid mechanism. Proc IMechE, Part I: J Systems and Control Engineering 2017; 231(9): 717–728.
13. Gao G, Ye M and Zhang M. Synchronous robust sliding mode control of a parallel robot for automobile electrocoating conveying. IEEE Access 2019; 7: 85838–85847.
14. Bengio Y. Learning deep architectures for AI. Found Trends Mach Learn 2009; 2(1): 1–127.
15. Yarotsky D. Error bounds for approximations with deep ReLU networks. Neural Netw 2017; 94: 103–114.
16. He K, Zhang X, Ren S, et al. Delving deep into rectifiers: surpassing human-level performance on ImageNet classification. In: IEEE International conference on computer vision (ICCV), Santiago, February 2015, pp.1026–1034.
17. Man Z, Wu H, Liu S, et al. A new adaptive backpropagation algorithm based on Lyapunov stability theory for neural networks. IEEE Trans Neural Netw 2006; 17(6): 1580–1591.