

Article

Trajectory Tracking Control of Fast Parallel SCARA Robots with Fuzzy Adaptive Iterative Learning Control for Repetitive Pick-and-Place Operations

Guanglei Wu  and Bin Niu * and Qiancheng Li

School of Mechanical Engineering, Dalian University of Technology, Dalian 116024, China; gwu@dlut.edu.cn (G.W.)

* Correspondence: niubin@dlut.edu.cn

Abstract: Aiming at enhanced suppression of external disturbances and high-precision trajectory tracking of parallel SCARA robot dedicating to fast pick-and-place operations, this work presents the integrated control design of iterative learning algorithm, adaptive control and fuzzy rules, namely, fuzzy adaptive iterative learning control, for such type of robots. A step-design approach is adopted to ensure the adaptability of the designed control law, which is reflected in two aspects: ① the feedback gain of the controller is adjusted by the fuzzy rules; ② the adaptive unknown parameters are obtained by means of iterative learning estimation to suppress the uncertainties and external disturbances. The stability of the designed controller is analyzed and proved by the Lyapunov theory, and the effectiveness is verified by observing the tracking errors in joint space along with the testing pick path, in comparison with different iterative learning based algorithms. After the first-iteration learning, the motion errors of the four actuated joints can be reduced by 56.5%, 45.8%, 46.4% and 39.8%, respectively, and after 15 iterations of learning control, the final angular errors by the designed control law converge to 0.7×10^{-4} degree maximally. The varying maximum, root-mean-squared and mean angular displacement errors of the actuation joints can converge to zero values with the increasing iterations rapidly, which shows the robustness, effectiveness and advantages of the designed control law. The designed control law can be generalized to high-speed parallel pick-and-place robot to ensure high-precision trajectory tracking for high-quality material handling tasks.

Keywords: trajectory tracking control; pick-and-place application; parallel robot; fuzzy control theory; iterative learning



Citation: Wu, G.; Niu, B.; Li, Q. Trajectory Tracking Control of Fast Parallel SCARA Robots with Fuzzy Adaptive Iterative Learning Control for Repetitive Pick-and-Place Operations. *Electronics* **2023**, *12*, 4995. <https://doi.org/10.3390/electronics12244995>

Academic Editor: Christos Volos

Received: 1 November 2023

Revised: 9 December 2023

Accepted: 12 December 2023

Published: 13 December 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Lightweight parallel robots featuring SCARA motion (the term “SACRA” means Selective Compliance Assembly Robot Arm, a.k.a., Schönflies motion, namely, three independent translations and one rotation around an axis of fixed direction) have been extensively deployed in production lines for material handling [1], which usually carry out repetitive tasks. Considering the continuous execution of identical trajectory tracking, this results in the accumulated motion errors in the motion process. If the previous problem cannot be handled effectively in time, the robot will eventually deviate from the expected trajectory, leading to the reduced work quality. Besides, the presence of uncertainties in the working environment will also decrease the control quality of the robot [2]. In this light, the key issue to improve the tracking accuracy lies in the avoidance of error accumulation and improved robustness of the system. The focus of this work is the control design for this type of robot.

The trajectory tracking control of a robot means that it is essential to design a reasonable control law for robotic joints with a determined trajectory, which makes the real end-effector position, velocity and accelerations consistent with the expected motion parameters [3].

The PID control has been commonly adopted for various robots since it was proposed, thanks to the simple control scheme and eased parameter tuning. On the other hand, the uncertainties cannot be readily handled in the use of this type of control law, and therefore, the PID control is usually designed in combination with other control algorithms to improve the tracking accuracy [4–10], e.g., feedforward [11], adaptive [12,13], robust [14], fuzzy control [15], etc. Besides, a number of different control algorithms have been proposed and applied to the parallel pick-and-place (PnP) robots, such as computed-torque control (CTC) [16] and sliding mode control (SMC) [17,18], but these heavily depends on the accurate dynamic model of the robots, which is a challenging task due to the highly nonlinear coupled system. Other robust control, such as H_2 [19] and H_∞ [20] controls, are suitable for noise signals with known power spectral density or for the condition of satisfying energy boundedness without enough information about the noise signal [21]. In light of the repetitive tasks of the family of parallel SCARA robots, an algorithm of iterative learning control (ILC) is well adapted with the repetitive characteristics, compared to the commonly used control laws.

The concept of iterative learning originated from the experienced learning knowledge and was proposed by Uchiyama in 1978 [22], which is free of feedback control to overcome the presence of disturbance and uncertainties in the control. Modifications and improvements to ILC algorithm have been made to enhance the trajectory tracking performance for the robots. As an open-loop control algorithm, ILC algorithms can be designed combining P, D and PD units of PID control, of which the architecture of the integrated control laws can be classified into open-, closed- and open-closed-loops. In Ref. [23], an open-loop PD-ILC is designed for Delta robots, which requires multiple iterations for convergence of tracking errors, leading to increased computational burden. Aiming at suppression of the longitudinal slippage disturbance, an open-closed-loop P-ILC algorithm is designed and applicable to a mobile robot [24]. It is difficult for the previous control law to be compatible with learning stability and feedback convergence, since the single proportion gain is simultaneously in charge of the feedback task of the control system and learning gain. In addition, optimization algorithm, such as the artificial bee colony algorithm, can also be adopted to optimize the initial parameters of the iterations, which can produce a better performance compared to the ILC with a constant gain [25]. To improve the control performance of nonlinear systems, a discrete high-order PID-ILC is proposed, which is complicated in the real control due to that it requires the numerous information of foregoing iterations rather than the current iteration [26]. Aiming to suppress aperiodic external disturbance, an ILC algorithm with forgetting factor (i.e., working as a filter) is proposed, which can effectively reduce the accumulated disturbance varying with the iterations and can make the tracking error converge to a certain range [27,28]. The fuzzy rules can be integrated to ILC algorithm for real-time tuning of control parameter to resist external disturbances in fewer iterations, compared to the classic PD-ILC [29]. Combining neural network, a high-order ILC is designed, consisting of two aspects, namely, neural network feedforward item and high-order P-type iterative feedback item, which corresponds to the suppression of periodic disturbance and the control stability [30]. In view of the present ILC-based control laws, an integrated fuzzy adaptive ILC algorithm is to be designed to discard the shortcomings of a single control law, for the enhancement of the trajectory tracking accuracy.

In this work, the adaptability of the combined ILC control law is reflected in two aspects: ① the feedback gain of the controller is adjusted by the fuzzy rule; ② the adaptive unknown parameters are obtained by means of iterative learning estimation to suppress the uncertainties and external disturbances caused by the unknown parameters to improve the trajectory tracking performance of the system. The stability of the controller is proved with the Lyapunov theory. The proposed algorithm was verified in comparison with the different ILC-based algorithms, of which the results show the robustness and effectiveness of the proposed algorithm, resulting in high tracking accuracy of the robot.

2. Pick-and-Place Robot and Dynamic Model

Figure 1 shows a parallel SCARA robot, which is composed of four identical revolute-revolute-parallelgram-revolute (i.e., RRRIR) linkages and a screw pair-based mobile platform. In each limb, the driving link is to provide the power to the parallelogram structure from the actuation unit of servo motor and reducer. The parallelogram, a.k.a., Π joint, is made up of carbon fiber, for a lightweight design as well as increased rigidity and stability of the robot structure. The corresponding parameterized CAD model is depicted in Figure 2.

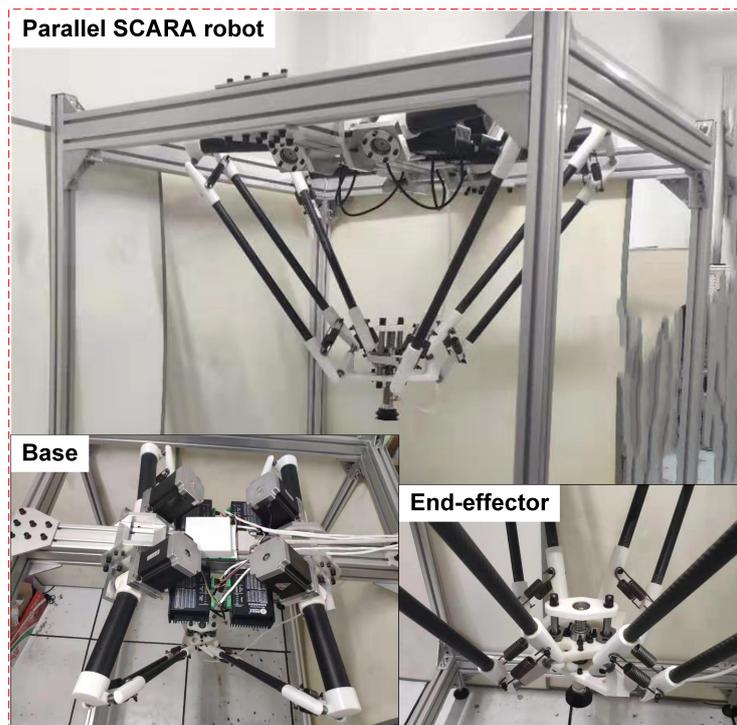


Figure 1. A parallel SCARA robot with four identical limbs and a screw pair-based mobile platform for pick-and-place application [31].

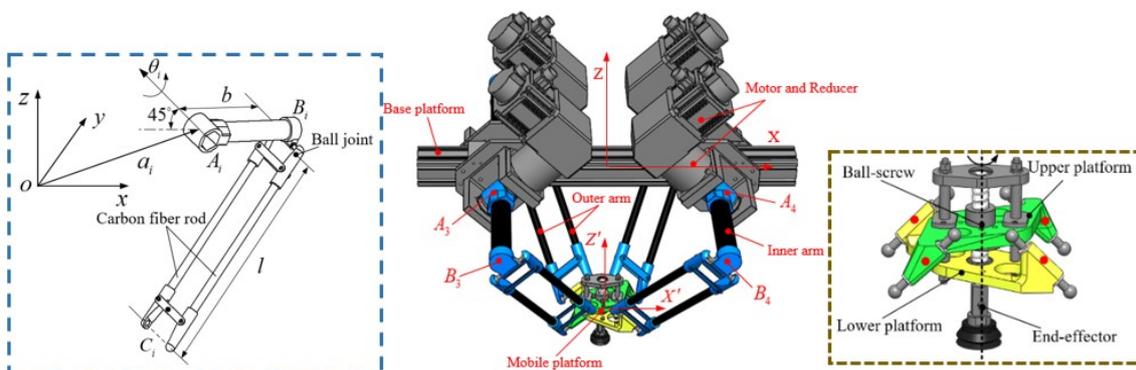


Figure 2. CAD model of the SCARA robot with parameterized linkage [32].

Prior to the control design of the robot, the dynamic model is to be derived, which can be expressed as:

$$\tau = M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + \omega(q, \dot{q}, t) \tag{1}$$

where $\tau \in \mathbb{R}^4$ represents a vector of the driving torque, $q = [\theta_1 \ \theta_2 \ \theta_3 \ \theta_4]^T$ represents the generalized coordinates, and \dot{q} , \ddot{q} denote the corresponding joint angular velocities and accelerations, respectively. The term $\omega(q, \dot{q}, t)$ represents the modeling error due to parameter uncertainties and external disturbance by changed environment. Moreover,

$\mathbf{M}(\mathbf{q})$, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ and $\mathbf{G}(\mathbf{q})$ stand for the inertia matrix, Coriolis and centrifugal term and gravity term, respectively, expressed as:

$$\mathbf{M}(\mathbf{q}) = \mathbf{J}_1^T \mathbf{M}_1 \mathbf{J}_1 + \mathbf{J}_2^T \mathbf{M}_2 \mathbf{J}_2 + \mathbf{I}_b \tag{2a}$$

$$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{J}_1^T \mathbf{M}_1 \dot{\mathbf{J}}_1 + \mathbf{J}_2^T \mathbf{M}_2 \dot{\mathbf{J}}_2 \tag{2b}$$

$$\mathbf{G}(\mathbf{q}) = -\left(\mathbf{J}_1^T \mathbf{M}_2 + \mathbf{J}_2^T \mathbf{M}_2 + \mathbf{M}_b\right) \mathbf{g} \tag{2c}$$

with:

$$\mathbf{M}_{1/2} = (m_{1/2} + 2m_l) \text{diag}[1 \quad 1 \quad 1 \quad 0] \tag{3a}$$

$$\mathbf{I}_b = \left(\frac{1}{3}m_b + m_l\right) b^2 \mathbf{I}_4 \tag{3b}$$

$$\mathbf{M}_b = \left(\frac{1}{2}m_b + m_l\right) \text{diag}[\cos \theta_1 \quad \cos \theta_2 \quad \cos \theta_3 \quad \cos \theta_4] \tag{3c}$$

$$\mathbf{J}_1 = [(\mathbf{c}_i - \mathbf{b}_i)^T \quad -\text{mod}(i + 1, 2) \frac{h}{2\pi} (\mathbf{c}_i - \mathbf{b}_i)^T \mathbf{k}] \tag{3d}$$

$$\mathbf{J}_2 = [(\mathbf{c}_i - \mathbf{b}_i)^T \quad -\text{mod}(i, 2) \frac{h}{2\pi} (\mathbf{c}_i - \mathbf{b}_i)^T \mathbf{k}] \tag{3e}$$

$$\mathbf{g} = [0 \quad 0 \quad -1 \quad 0]^T \tag{3f}$$

where \mathbf{J}_1 and \mathbf{J}_2 represent the Jacobian matrices mapping the velocities between the active joints and the upper/lower sub-platforms, respectively, \mathbf{b}_i and \mathbf{c}_i represent the Cartesian coordinates of points B_i and C_i in the reference frame and \mathbf{k} denotes the unit vector of z-axis. Moreover, \mathbf{I}_4 is a four-dimensional identity matrix. Skipping the details, the kinematic and dynamic models have been well documented in the previous works [2,31], together with the link dimensions and mass properties listed in Table 1.

Table 1. Geometric and mass parameters of the parallel robot.

Parameters	Meaning	Value	Unit
(a_x, a_y)	x-/y-coordinates of joint position	(286, 132)	mm
b/l	lengths of proximal/distal links	300/600	mm
r	radius of moving platform	100	mm
h	pitch of lead screw	10	mm
m_b/m_l	masses of proximal/distal links	0.5101/0.2670	kg
m_1/m_2	masses of upper/lower sub-platforms	0.9789/0.8914	kg

3. Fuzzy Adaptive Iterative Learning Control Design

In this section, the controller design for the robot under study is accomplished by the integrated design of ILC and adaptive control law as well as fuzzy rules.

3.1. Design of the Iterative Learning Control Law

Considering the nonlinear time-varying robotic system with repetitive work over a finite interval time $t \in [0, T]$, the following reasonable assumptions can be made:

Assumption 1. The robotic system can meet the boundary condition, i.e., $\mathbf{q}_k(0) = \mathbf{q}_d(0)$, $\dot{\mathbf{q}}_k(0) = \dot{\mathbf{q}}_d(0)$, where \mathbf{q}_k and $\mathbf{q}_d(0)$ stand for the real and expected trajectories, respectively, and k depicts the iteration number in the control law.

Assumption 2. The external disturbance $\boldsymbol{\tau}(\mathbf{q}, \dot{\mathbf{q}}, t)$ of the robot is bounded, which can meet an inequality $\sup \|\boldsymbol{\tau}(\mathbf{q}, \dot{\mathbf{q}}, t)\| \leq l$, with l being a positive constant.

For a general robotic manipulator, the dynamic system can have the following three properties:

Property 1. For the bounded and positive-definite inertia matrix $\mathbf{M}(\mathbf{q}_k, t)$ of a robot, there exists the constants δ and ζ to meet the inequality:

$$0 < \delta < \|\mathbf{M}(\mathbf{q}_k, t)\| < \zeta, 0 < \delta < \zeta \tag{4}$$

Property 2. Since the inertia matrix can meet the global Lipschitz condition [33], there is a positive constant L to hold the following inequality:

$$\|\mathbf{M}(\mathbf{q}_k, t) - \mathbf{M}(\mathbf{q}_{k-1}, t)\| \leq L\|\mathbf{q}_k(t) - \mathbf{q}_{k-1}(t)\| \tag{5}$$

Property 3. The Coriolis, centrifugal and gravitational terms of the dynamic equation can meet the condition of $\mathbf{C}(\mathbf{q}_k, \dot{\mathbf{q}}_k)\dot{\mathbf{q}}_k + \mathbf{G}(\mathbf{q}_k) = \mathbf{\Phi}(\mathbf{q}_k, \dot{\mathbf{q}}_k)\boldsymbol{\gamma}_k(t)$, where $\mathbf{\Phi}(\mathbf{q}_k, \dot{\mathbf{q}}_k) \in \mathbf{R}^{n \times m}$ represents the regression matrix to depict the motions of robotic joints and $\boldsymbol{\gamma}_k(t) \in \mathbf{R}^{m \times 1}$ is a vector of unknown parameters with regard to the robot.

With the aforementioned conditions and constraints, an iterative learning algorithm is able to be designed to make full use of the effective information stored in the system for learning, which can ensure that the output variables converge to the bounded threshold of desired values, expressed as [34]:

$$\boldsymbol{\tau}_{k+1}(t) = \boldsymbol{\tau}_k(t) + \boldsymbol{\tau}_{fore} + \boldsymbol{\tau}_{back} \tag{6}$$

where $\boldsymbol{\tau}$ represents the input torque and $\boldsymbol{\tau}_{fore}$, $\boldsymbol{\tau}_{back}$ stand for the feedforward and feedback compensation torques, respectively, which take the following form:

$$\boldsymbol{\tau}_{fore} = \mathbf{K}_p \mathbf{e}_k(t) + \mathbf{K}_d \dot{\mathbf{e}}_k(t) \tag{7a}$$

$$\boldsymbol{\tau}_{back} = (1 - \alpha)\mathbf{K}_p \mathbf{e}_{k+1}(t) + (1 - \beta)\mathbf{K}_d \dot{\mathbf{e}}_{k+1}(t) \tag{7b}$$

where \mathbf{K}_p and \mathbf{K}_d are, respectively, diagonal positive gain matrices for proportion and differential PD parameters in the feedforward control, $\mathbf{e}_k(t)$ and $\dot{\mathbf{e}}_k(t)$ represent the joint errors in terms of angular displacements and velocities, respectively, in the k th iteration and α and β stand for the gain coefficients.

3.2. Fuzzy Controller Design

In the control design, the fuzzy rule is adopted to adaptively adjust the gain of the controller for the improved suppression of the disturbance of the system. A particular control algorithm is formulated as:

$$\mathbf{K}_p = \mathbf{K}_{p0} + \Delta\mathbf{K}_p \tag{8a}$$

$$\mathbf{K}_d = \mathbf{K}_{d0} + \Delta\mathbf{K}_d \tag{8b}$$

where \mathbf{K}_{p0} and \mathbf{K}_{d0} represent the initial values of the gain and $\Delta\mathbf{K}_p$, $\Delta\mathbf{K}_d$ are the corresponding compensation values of the fuzzy control output.

Let us define a fuzzy set with seven fuzzy subsets, i.e., $\{NB, NM, NS, ZO, PS, PM, PB\}$, which, respectively, stand for Negative Big, Negative Medium, Negative Small, Zero, Positive Small, Positive Medium and Positive Big in the fuzzy control language [35]. Taking the motion errors of angular displacement \mathbf{e} and velocity $\dot{\mathbf{e}}$ of the actuated joints as the input variables of the controller, the physical domains of angular displacement \mathbf{e} and velocity $\dot{\mathbf{e}}$ errors of joints 1 and 3 are, respectively, set to $\{-0.05, 0.05\}$ and $\{-0.01, 0.05\}$, while the physical domains of motion errors \mathbf{e} and $\dot{\mathbf{e}}$ for joints 2 and 4 are, respectively, set to $\{-0.01, 0.01\}$ and $\{-0.01, 0.05\}$, in accordance with the parameter tuning determined in previous research [32]. On the other hand, the controller gain compensations $\Delta\mathbf{K}_p$ and $\Delta\mathbf{K}_d$ are used as output variables, of which the ranges are set to $\{-500, 2000\}$ and $\{-80, 120\}$, respectively. The fuzzy domains are defined in its generalized form, for which the values are set to $\{-3, 3\}$. Moreover, the physical and fuzzy domains are mapped by means of

scaling factors. To this end, the membership functions for the input and output variables of the controller are depicted in Figure 3. Here, the triangular membership function is adopted to ease the control algorithm for fast computation, while Z/S-shaped membership functions are used at the boundaries of the output parameters to realize the parameter regulation for enhanced adaptability of the controller.

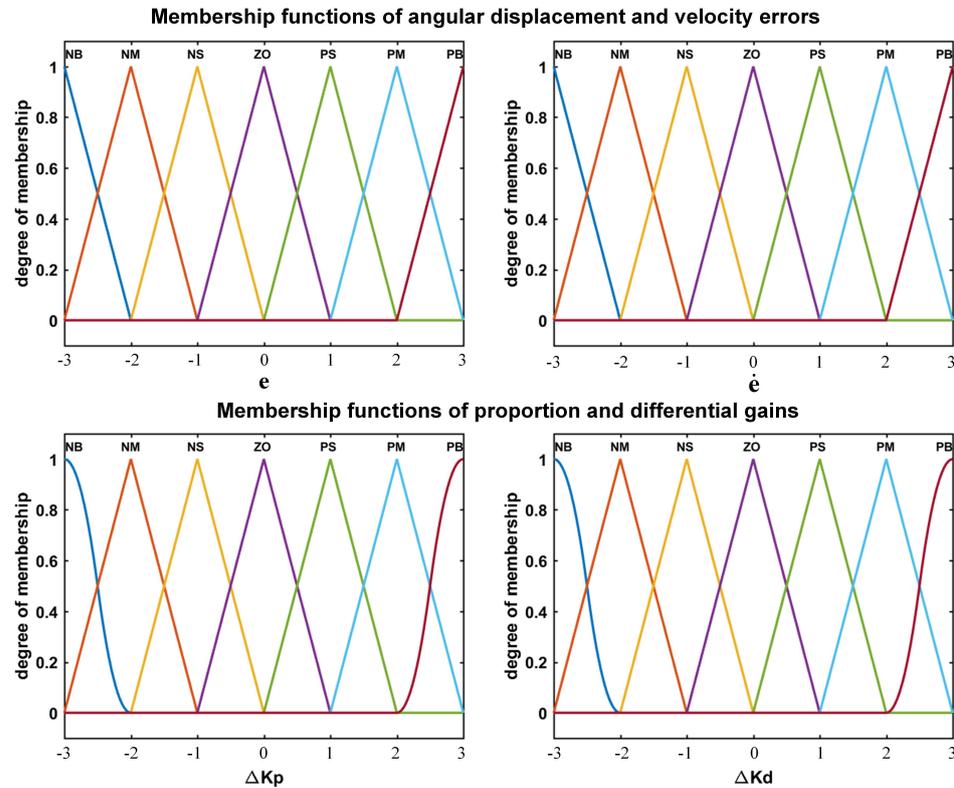


Figure 3. The membership functions for the input and output variables of the fuzzy controller.

When the robot moves with angular displacement and velocity errors, which offsets from the expected trajectory, the fuzzy subset of gain compensations can be obtained from the fuzzy rule table. Based upon the analysis of the influence of proportional and differential gains onto the performance of the controller, the fuzzy strategies are summarized in Tables 2 and 3, respectively. After the operation of defuzzification, the surfaces of varying proportional and differential gains of fuzzy algorithm for joint 1 is plotted in Figure 4, and the surfaces of the remaining joints have the similar distributions over different physical domains as previously mentioned.

Table 2. Fuzzy rule table of proportional gain tuning.

ΔK_p	de						
e	NB	NM	NS	ZO	PS	PM	PB
NB	PB	PB	PM	PM	PS	PS	ZO
NM	PB	PB	PM	PM	PS	ZO	ZO
NS	PM	PM	PM	PS	ZO	NS	NM
ZO	PM	PS	PS	ZO	NS	NM	NM
PS	PS	PS	ZO	NS	NS	NM	NM
PM	ZO	ZO	NS	NM	NM	NM	NB
PB	ZO	NS	NS	NM	NM	NB	NB

Table 3. Fuzzy rule table of differential gain tuning.

ΔK_D	$d\dot{e}$							
	\dot{e}	NB	NM	NS	ZO	PS	PM	PB
NB	NB	NB	NM	NM	NS	ZO	ZO	ZO
NM	NB	NB	NM	NM	NS	ZO	ZO	ZO
NS	NM	NM	NM	NS	ZO	PS	PM	PM
ZO	NM	NS	NS	ZO	PS	PM	PM	PM
PS	NS	NS	ZO	PS	PS	PM	PM	PM
PM	ZO	ZO	PS	PM	PM	PM	PM	PB
PB	ZO	ZO	PS	PM	PM	PB	PB	PB

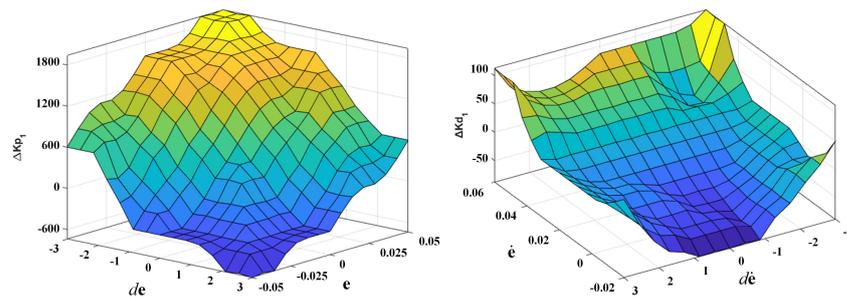


Figure 4. Surface of varying proportional and differential gains of fuzzy controller for joint 1.

3.3. Adaptive Iterative Learning Law

The use of the ILC algorithm is to learn the useful information from previous movements to correct the current state, which can effectively suppress the repetitive uncertain disturbances. On the other hand, there exists some non-repetitive external disturbances in a real situation. In order to make the robot adaptive to the external environment and to improve the trajectory tracking precision, an adaptive [36] ILC law is designed below:

$$\tau_k(t) = \mathbf{K}_p \mathbf{e}_k(t) + \mathbf{K}_d \dot{\mathbf{e}}_k(t) + \boldsymbol{\zeta}(\mathbf{q}_k, \dot{\mathbf{q}}_k, \mathbf{y}_k) \gamma_k(t) + \boldsymbol{\omega}_k(t) + \mathbf{D} \text{sat}(\mathbf{y}_k(t)) \tag{9}$$

with:

$$\gamma_k(t) = \gamma_{k-1}(t) + \alpha \boldsymbol{\zeta}^T(\mathbf{q}_k, \dot{\mathbf{q}}_k, \mathbf{y}_k) \mathbf{y}_k(t) \tag{10}$$

where $\{\mathbf{K}_p, \mathbf{K}_d\} \in \mathbb{R}^n$ are the gain matrices corresponding to the proportion and differential parts in PD control and $\mathbf{e}_k(t) = \mathbf{q}_d(t) - \mathbf{q}_k(t)$, $\dot{\mathbf{e}}_k(t) = \dot{\mathbf{q}}_d(t) - \dot{\mathbf{q}}_k(t)$ represent the motion errors of the actuated joints in terms of angular displacement and velocity, respectively. For the vector of unknown parameters of the robot $\gamma_k(t)$, $\gamma_{k-1}(t) = \mathbf{0}$ exists. The generalized error is expressed as $\mathbf{y}_k(t) = \dot{\mathbf{e}}_k(t) + \Lambda \mathbf{e}_k(t)$, Λ being a positive constant. The expression $\|\boldsymbol{\omega}_{k+1}^{(l)}(t) - \boldsymbol{\omega}_k^{(l)}(t)\| \leq \mathbf{D}$ exists, where $\boldsymbol{\omega}^{(l)}$ depicts the repetitive external disturbance. Let us define $\boldsymbol{\zeta}^T(\mathbf{q}_k, \dot{\mathbf{q}}_k, \mathbf{y}_k) = [\phi(\mathbf{q}_k, \dot{\mathbf{q}}_k) \text{ sat}(\mathbf{y}_k)]$, with $\text{sat}(\mathbf{y}_k)$ being the saturation form of the generalized error, which can decrease the influence of discontinuous torques caused by the switching function, to ensure the controller have continuous output.

3.4. Integration of Fuzzy Adaptive ILC

On the basis of previously designed control laws, the integrated control scheme is designed as depicted in Figure 5, which consists of three major control laws, i.e., adaptive control of parameter tuning, adaptive feedback control of gains and compensation robust control of external disturbances. The adaptive parameters are updated in the time and iterative domains simultaneously, where the parameter correction term includes a partially linearized dynamic model of the robot and a saturated generalized error. The design of the generalized error as a saturated function is able to learn from the displacement and velocity errors, to reduce the joint buffetings in the function switching. Moreover, the fuzzy controller for the adaptive feedback of gain can work to adjust PD gains in real time, which

It is known that $V_k(\mathbf{q}_k(0), \mathbf{e}_k(0), \dot{\mathbf{e}}_k(0)) = 0$, and thus:

$$\begin{aligned} \Delta W_k &= V_k - V_{k-1} - \frac{1}{2\alpha} \int_0^t (\bar{\zeta}_k^T \bar{\zeta} + 2\bar{\zeta}_{k-1}^T \bar{\zeta}_{k-1}) d\tau \\ &\leq -V_{k-1} + \int_0^t [\dot{\mathbf{e}}_k^T \boldsymbol{\zeta}(\mathbf{q}_k, \dot{\mathbf{q}}_k, \mathbf{y}_k) \tilde{\mathbf{q}}_k(t) - \dot{\mathbf{e}}_k^T \mathbf{K}_d \dot{\mathbf{e}}_k] d\tau \\ &\quad - \frac{\alpha}{2} \int_0^t [\mathbf{y}_k^T(t) \boldsymbol{\zeta}(\mathbf{q}_k, \dot{\mathbf{q}}_k, \mathbf{y}_k) \boldsymbol{\zeta}^T(\mathbf{q}_k, \dot{\mathbf{q}}_k, \mathbf{y}_k) \mathbf{y}_k(t)] d\tau \\ &\quad - \int_0^t [\mathbf{y}_k^T(t) \boldsymbol{\zeta}(\mathbf{q}_k, \dot{\mathbf{q}}_k, \mathbf{y}_k) \tilde{\mathbf{q}}_k(t)] d\tau \end{aligned} \tag{16}$$

Substituting the generalized errors $\mathbf{y}_k(t) = \dot{\mathbf{e}}_k(t) + \Lambda \mathbf{e}_k(t)$ into the previous equation leads to:

$$\begin{aligned} \Delta W_k &\leq -V_{k-1} - \frac{\alpha}{2} \int_0^t [\mathbf{y}_k^T(t) \boldsymbol{\zeta}(\mathbf{q}_k, \dot{\mathbf{q}}_k, \mathbf{y}_k) \boldsymbol{\zeta}^T(\mathbf{q}_k, \dot{\mathbf{q}}_k, \mathbf{y}_k) \mathbf{y}_k(t)] d\tau \\ &\quad - \int_0^t \dot{\mathbf{e}}_k^T \mathbf{K}_d \dot{\mathbf{e}}_k d\tau - \int_0^t [\Lambda \mathbf{e}_k^T \boldsymbol{\zeta}(\mathbf{q}_k, \dot{\mathbf{q}}_k, \mathbf{y}_k) \tilde{\mathbf{q}}_k(t)] d\tau \leq 0 \end{aligned} \tag{17}$$

from which it turns out that the energy increment ΔW_k is a non-positive value with the increasing iterations, and thus, W_k with the increasing iteration constitute a decreasing sequence, namely, a non-incremental function.

(2) Proof of boundedness and convergence of motion errors

The next step is to prove that W_0 is bounded, and considering $k = 0$, the differentiation of Equation (11) results in:

$$\dot{W}_0 = \dot{V}_k(0)(\mathbf{q}_k(0), \mathbf{e}_k(0), \dot{\mathbf{e}}_k(0)) + \frac{1}{2\alpha} \tilde{\zeta}_k(\tau) \tilde{\zeta}(\tau) \tag{18}$$

Combining Equation (15) with the foregoing expression yields:

$$\begin{aligned} \dot{W}_0 &\leq \dot{\mathbf{e}}_0^T [\boldsymbol{\zeta}(\mathbf{q}_0, \dot{\mathbf{q}}_0, \mathbf{y}_0) \tilde{\mathbf{q}}_0 - \mathbf{K}_d \dot{\mathbf{e}}_0 - \mathbf{D}_{sat}(\mathbf{y}_0)] + \frac{1}{2\alpha} \tilde{\zeta}_0^T \tilde{\zeta}_0 \\ &\leq \dot{\mathbf{e}}_0^T [\boldsymbol{\zeta}(\mathbf{q}_0, \dot{\mathbf{q}}_0, \mathbf{y}_0) \tilde{\mathbf{q}}_0 - \mathbf{K}_d \dot{\mathbf{e}}_0] + \frac{1}{2\alpha} \tilde{\zeta}_0^T \tilde{\zeta}_0 \end{aligned} \tag{19}$$

Due to the presence of $\hat{\mathbf{q}}_{-1}(t) = 0$, in accordance with Equation (10), the following equation exists:

$$\gamma_0(t) = \alpha \boldsymbol{\zeta}(\mathbf{q}_0, \dot{\mathbf{q}}_0, \mathbf{y}_0) \mathbf{y}_0(t) \tag{20}$$

which can be substituted into Equation (21):

$$\dot{W}_0 \leq -\dot{\mathbf{e}}_0^T \mathbf{K}_d \dot{\mathbf{e}}_0 + \frac{1}{\alpha} \gamma_0^T \tilde{\zeta}_0 + \frac{1}{2\alpha} \tilde{\zeta}_0^T \tilde{\zeta}_0 \leq -\dot{\mathbf{e}}_0^T \mathbf{K}_d \dot{\mathbf{e}}_0 - \frac{1}{2} \tilde{\zeta}_0^T \alpha^{-1} \tilde{\zeta}_0 + \zeta^T \alpha^{-1} \tilde{\zeta}_0 \tag{21}$$

Based upon Young inequality [38], the following can be obtained:

$$\zeta^T \alpha^{-1} \tilde{\zeta}_0 \leq \eta \|\alpha^{-1} \tilde{\zeta}_0\| + \frac{1}{4\eta} \|\zeta\|^2 \tag{22}$$

Consequently, the following inequality can be deduced:

$$\dot{W}_0 \leq - \left[\frac{1}{2} \lambda_{\min}(\alpha^{-1}) - \eta \lambda_{\max}(\alpha^{-1}) \right] \|\tilde{\zeta}_0\|^2 - \lambda_{\min}(\mathbf{K}_d) \|\dot{\mathbf{e}}_0\|^2 + \frac{1}{4\eta} \|\zeta\|^2 \tag{23}$$

with:

$$\eta \leq \frac{\lambda_{\min}(\alpha^{-1})}{2\eta \lambda_{\max}^2(\alpha^{-1})} \tag{24}$$

where $\lambda_{\min}(\cdot)$ and $\lambda_{\max}(\cdot)$ stand for the minimum and maximum eigenvalues of (\cdot) , respectively. Since $\|\dot{\mathbf{e}}_0\|$, $\|\tilde{\zeta}_0\|$ and $\|\zeta\|$ are bounded in the time domain $[0, T]$, W_0 is bounded, from which the following condition can be derived:

$$\lim_{k \rightarrow \infty} e_k(t) = 0 \quad \& \quad \lim_{k \rightarrow \infty} \dot{e}_k(t) = 0, \quad \forall t \in [0, T] \tag{25}$$

The foregoing equation implies that the motion errors have the characteristics of iterative convergence, and when the expected errors do not vary with the iterations, the errors converge to null.

To this end, the stability of the designed controller is analyzed and proved.

4. Results and Discussions of the Control Design

In order to verify the effectiveness of the designed control algorithm, the SCARA parallel robot (see Figure 2) is tested along with a PnP trajectory with the dimension of $25 \times 305 \times 25$ mm, which is written as a B-spline curve [39] for trajectory smoothing. Figure 6 shows the testing trajectory as well as the motion profiles of the end-effector, and the traveling time is 0.25 s, which means that the working frequency is 120 CPM (i.e., cycles per minute [40]).

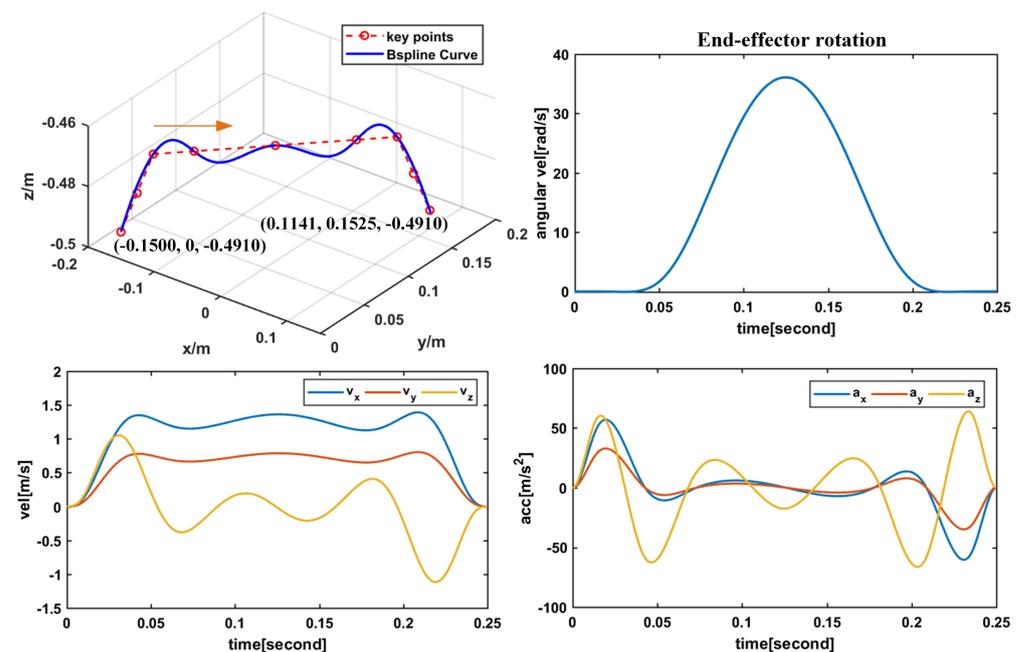


Figure 6. The motion profiles of the robot end-effector for trajectory tracking.

To simulate the designed control algorithm, a control platform is built with the Matlab/Simscape of version 2020a, which consists of five modules in terms of trajectory planning, control law of fuzzy AIL, external disturbance, robot system and sensor measurement, as depicted in Figure 7. The input motion parameters in the module of “trajectory generation” are calculated from the inverse kinematics according to the tracked trajectory, as displayed in Figure 6. The designed control algorithm depicted in Figure 5 is embodied in the “fuzzy-AIL control” block, and the external disturbance is managed with “Flag_Ft” function as the Switch to be included in the simulation or not. The multi-body system in the “Parallel Robot” module is generated and imported from Solidworks™ 2016 through the Matlab-Solidworks interface addon “Simscape Multibody Link” for mechanics simulation, and the torque feedforward control is adopted as the close-loop control law based on the dynamic model, as depicted in Figure 8. The “measurement” module outputs the sensor data attached to the joints in the Simscape package.

In the previous work [2], torque feedforward control was designed and the PD parameters were optimized by minimizing the trajectory tracking errors. Comparing the tracking accuracy with different local optimums of the control parameters, the control parameters of the gains are selected as:

$$\mathbf{K}_p = \text{diag}[2.4 \ 2.4 \ 2.4 \ 2.4] \times 10^4, \mathbf{K}_d = \text{diag}[1.2 \ 1.2 \ 1.2 \ 1.2] \times 10^2 \quad (26)$$

and the adaptive parameters are set to:

$$\alpha = \text{diag}[82 \ 82 \ 82 \ 82 \ 82], \Lambda = 340 \tag{27}$$

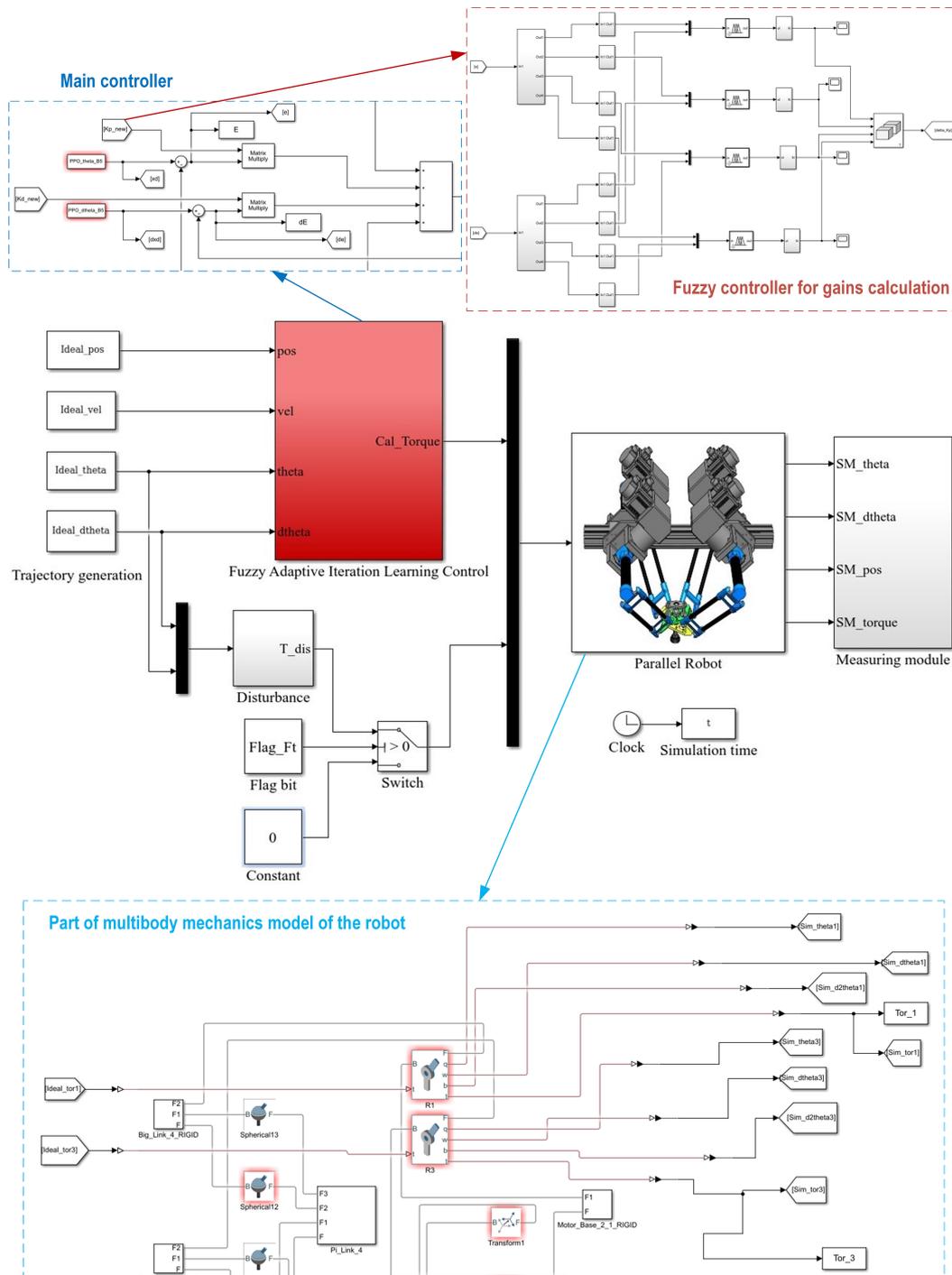


Figure 7. Simulation platform of the fuzzy adaptive iterative learning control algorithm.

When the robot tracks the planned path in Figure 6, the sampling time and the number of iterations are set to 1 ms and 15, respectively. Here, three indices, the maximum (E_{max}), mean (E_{ave}) and root-mean-squared (E_{rms}) angular errors of the joints, are defined for the evaluation of the tracking accuracy. Figure 9 depicts the varying angular errors of the joints with respect to the times from the first to final iterations, from which it can be seen that the angular errors of all the joints in the initial iteration, are large and converge to very

small values in the last iteration. The varying motion errors verify the convergence of the designed control law with the increasing iterations.

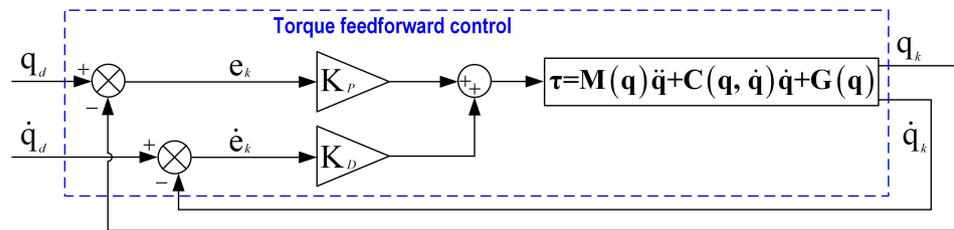


Figure 8. The closed-loop torque-based feedforward PD control diagram.

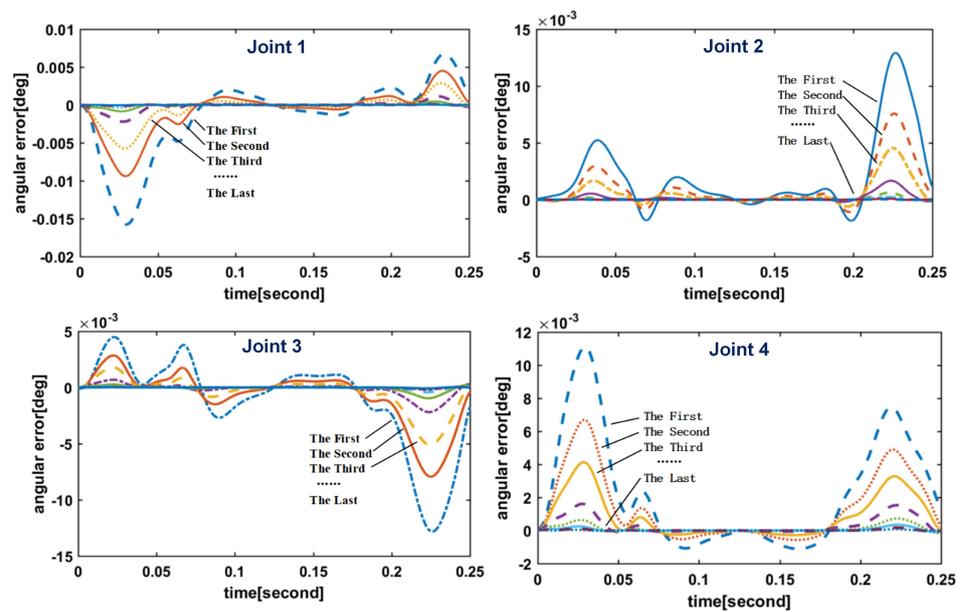


Figure 9. The varying angular errors of the actuated joints with the iterations.

Moreover, the convergence curves of the angular errors E_{rms} for all joints varying with the iterations are given in Figure 10. It can be observed that in the first iteration without controller learning to store the previously information, the errors of angular displacements are relatively large. After the learning of first iteration, the errors of each joint can be reduced by 56.5%, 45.8%, 46.4% and 39.8%, respectively. With the iterations increasing, all the angular errors of joints gradually become smaller; particularly, the errors significantly decrease in the first five iterations. The convergence curves of E_{rms} errors of all the joints turn to be stable after the ninth iteration, and finally almost converge to zero. The aforementioned evaluation indices of angular displacement errors are listed in Table 4, which are close to zero, which shows the effectiveness of error convergence by the designed controller.

To verify the robustness of the control system, periodic $\tau_{dis_re} = k \sin(\lambda t + \varphi)$ and non-periodic disturbance $\tau_{dis} = 2 \sin(q_d) - \sin(\dot{q}_d)$ are designed to simulate the external disturbances that can affect the tracking accuracy, such as the assembly errors, joint friction, unpredictable external environment during the production process, etc. The trajectory tracking errors of angular displacements with/without disturbances in the final iteration are depicted in Figure 11, from which the comparison shows that the two curves of the trajectory tracking errors are almost consistent. This means that even if there is an external disturbance, the designed controller can make the robot track the expected trajectory with ignorable errors, which implies that the designed control algorithm can well suppress various periodic and non-periodic disturbances with good robustness. Besides, the angular velocity errors of the actuated joints are shown in Figure 12, from which it can be seen that

the errors are very small and smooth except at the start and end of the trajectory with the higher accelerations.

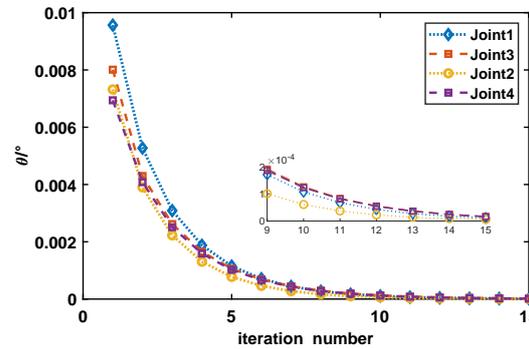


Figure 10. The convergence of the angular errors E_{rms} of the joints with the increasing iterations.

Table 4. The final converged angular displacement errors by the designed control law.

Joint No.	E_{max}	E_{ave}	E_{rms}
1	4.0×10^{-5}	6.4×10^{-6}	1.1×10^{-5}
2	2.5×10^{-5}	3.2×10^{-6}	5.7×10^{-5}
3	7.0×10^{-5}	7.4×10^{-6}	1.6×10^{-5}
4	6.4×10^{-5}	8.4×10^{-6}	1.6×10^{-5}

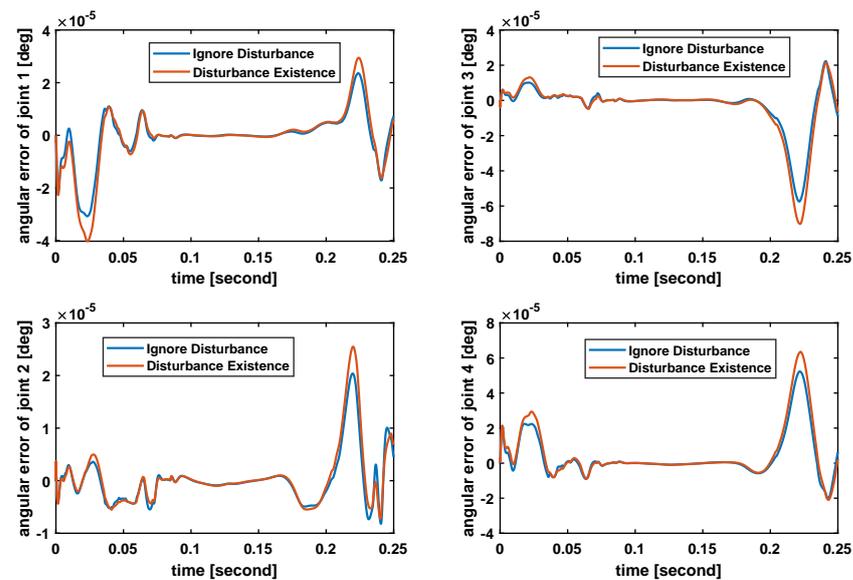


Figure 11. The angular errors of the actuated joints in the final iteration.

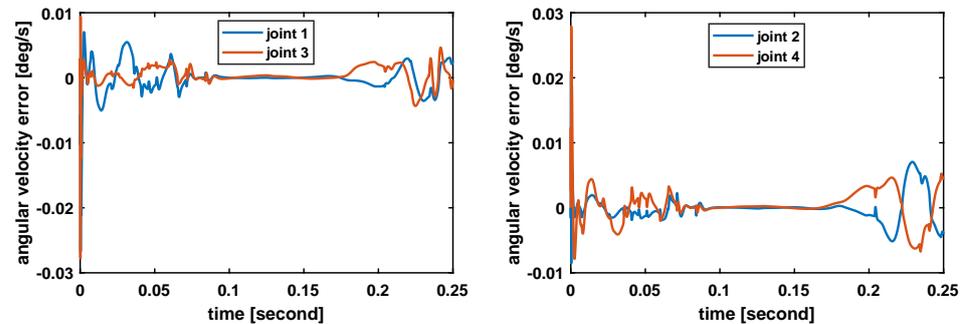


Figure 12. The angular velocity errors of the actuated joints in the final iteration.

In order to show the advantage of the designed fuzzy adaptive ILC algorithm, a comparative study is carried out with the classical D-type and PD-type ILC algorithms.

Figure 13 shows the comparison of the maximum initial and final converged angular errors of all the joints. It can be seen that the maximum initial angular errors generated by the D-type and PD-type ILC algorithms are equal to 0.87 and 0.61 degrees [34], respectively, which are much larger than that of fuzzy adaptive ILC, where the maximum initial angular error is equal to 0.028 degrees. Moreover, after 15 iterations of learning control, the final angular errors by the three control laws converge to 1.9×10^{-4} , 1.5×10^{-4} and 0.7×10^{-4} degree maximally, respectively, which shows that the designed fuzzy adaptive ILC algorithm can speed up the convergence in fewer iterations and can make the trajectory tracking errors converge to smaller values. Compared to other ILC-based control laws, the fuzzy adaptive ILC designed in this work can ensure high-precision trajectory tracking for high-quality material handling tasks.

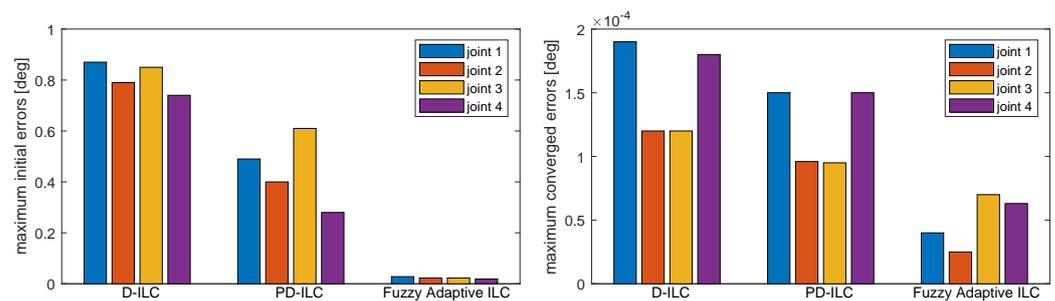


Figure 13. Comparison of the maximum initial and final converged angular errors with different ILC-based control laws.

It is noteworthy that the previous implementation of simulation is carried out with 64-bit Windows 7 operation system, of which the processor is an Intel i5-9400f with 8 GB of RAM. The computational time of the designed control algorithm in each iteration is about 0.3–0.5 s, and thus, the optimal control parameters can be converged in seconds, which implies that the presented control law is suitable for stable repetitive pick-and-place tasks.

5. Conclusions

This paper presents the integrated control design of iterative learning algorithm, fuzzy rules and adaptive control, i.e., fuzzy adaptive ILC, for the parallel SCARA robot, aiming to improve the trajectory tracking accuracy for high-quality material handling tasks. A step-design approach is adopted to ensure the adaptability of the designed control law that is reflected in two aspects: ① the feedback gain of the controller is adjusted by the fuzzy rules; ② the adaptive unknown parameters are obtained by means of iterative learning estimation to suppress the uncertainties and external disturbances caused by the unknown parameters for high-precision trajectory tracking of the robot. The stability of the controller is analyzed and proved with the Lyapunov theory.

The effectiveness of the designed control algorithm is verified by observing the tracking errors in joint space, where the robot tracks along with a PnP testing trajectory. It turns out that the varying maximum, root-mean-squared and mean angular displacement errors of the actuation joints converge to almost zero values with the increasing iterations rapidly. Moreover, two kinds of external disturbances, i.e., periodic and non-periodic ones, are designed to simulate the external working environment of the robot, from which the results show that the designed control scheme can well suppress various disturbances with good robustness. Compared to different ILC-based control laws, the designed fuzzy adaptive ILC algorithm in this work can speed up the convergence of tracking errors to smaller values, which can ensure high-precision trajectory tracking. The presented results show the robustness, effectiveness and advantages of the designed control law, which can be generalized to the high-speed parallel PnP robots of this family.

Author Contributions: Conceptualization, methodology, data curation, writing—original draft preparation and editing, supervision, G.W.; conceptualization, funding acquisition, writing—review and editing, B.N.; software, validation, formal analysis, Q.L. All authors have read and agreed to the published version of the manuscript.

Funding: The support from the Fundamental Research Funds for the Central Universities (No. DUT22QN222) is greatly appreciated.

Data Availability Statement: Data available upon the requests from the authors.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Wu, G.; Shen, H. *Parallel PnP Robots*; Springer: Singapore, 2021; pp. 1–15.
2. Lin, Z.; Cui, C.; Wu, W. Dynamic modeling and torque feedforward based optimal fuzzy PD control of a high-speed parallel manipulator. *J. Robot. Control* **2021**, *2*, 527–538. [[CrossRef](#)]
3. Siciliano, B.; Sciavicco, L.; Villani, L.; Oriolo, G. Trajectory Planning. In *Robotics: Modelling, Planning and Control*; Springer: London, UK, 2008; pp. 161–189.
4. Zubizarreta, A.; Marcos, M.; Cabanes, I.; Pinto, C. A procedure to evaluate extended computed torque control configurations in the Stewart–Gough platform. *Robot. Auton. Syst.* **2011**, *59*, 770–781. [[CrossRef](#)]
5. Codourey, A. Dynamic modeling of parallel robots for computed-torque control implementation. *Int. J. Robot. Res.* **1998**, *17*, 1325–1336. [[CrossRef](#)]
6. Craig, J.J.; Hsu, P.; Sastry, S.S. Adaptive control of mechanical manipulators. *Int. J. Robot. Res.* **1987**, *6*, 16–28. [[CrossRef](#)]
7. Chen, Y.; Ma, G.; Lin, S.; Gao, J. Adaptive fuzzy computed-torque control for robot manipulator with uncertain dynamics. *Int. J. Adv. Robot. Syst.* **2012**, *9*, 237. [[CrossRef](#)]
8. Hyon, S.; Suewaka, D.; Torii, Y.; Oku, N. Design and experimental evaluation of a fast torque-controlled hydraulic humanoid robot. *IEEE/ASME Trans. Mechatron.* **2017**, *22*, 623–634. [[CrossRef](#)]
9. Koessler, A.; Bouton, N.; Briot, S.; Bouzgarrou, B.C.; Mezouar, Y. Linear adaptive computed torque control for singularity crossing of parallel robots. In *ROMANSY 22—Robot Design, Dynamics and Control*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 222–229.
10. Qi, Z.; McInroy, J.E.; Jafari, F. Trajectory tracking with parallel robots using low chattering, fuzzy sliding mode controller. *J. Intell. Robot. Syst.* **2007**, *48*, 333–356. [[CrossRef](#)]
11. Choi, H.; Konno, A.; Uchiyama, M. Design, implementation, and performance evaluation of a 4-DOF parallel robot. *Robotica* **2010**, *28*, 107–118. [[CrossRef](#)]
12. Azad, F.A.; Rahimi, S.; Hairi Yazdi, M.R.; Masouleh, M.T. Design and Evaluation of Adaptive and Sliding Mode Control for a 3-DOF Delta Parallel Robot. In Proceedings of the 2020 28th Iranian Conference on Electrical Engineering (ICEE), Tabriz, Iran, 4–6 August 2020; pp. 1–7.
13. Bae, B.; Lee, D.-H. Design of a Four-Wheel Steering Mobile Robot Platform and Adaptive Steering Control for Manual Operation. *Electronics* **2023**, *12*, 3511. [[CrossRef](#)]
14. Khosravi, M.A.; Taghirad, H.D. Robust PID control of fully-constrained cable driven parallel robots. *Mechatronics* **2014**, *24*, 87–97. [[CrossRef](#)]
15. Tiep, D.K.; Lee, K.; Im, D.-Y.; Ryoo, Y.K. Design of Fuzzy-PID Controller for Path Tracking of Mobile Robot with Differential Drive. *Int. J. Fuzzy Logic Intell. Syst.* **2018**, *18*, 220–228. [[CrossRef](#)]
16. Vivas, A.; Poignet, P. Predictive functional control of a parallel robot. *Control Eng. Pract.* **2005**, *13*, 863–874.
17. Jia, H.; Shang, W.; Xie, F.; Zhang, B.; Cong, S. Second-order sliding-mode-based synchronization control of cable-driven parallel robots. *IEEE/ASME Trans. Mechatron.* **2020**, *25*, 383–394. [[CrossRef](#)]
18. Sachan, S.; Swarnkar, P. Intelligent Fractional Order Sliding Mode Based Control for Surgical Robot Manipulator. *Electronics* **2023**, *12*, 729. [[CrossRef](#)]
19. Rotea, M.A. The generalized H₂ control problem. *Automatica* **1993**, *29*, 373–385. [[CrossRef](#)]
20. Xu, S.Y.; Chen, T.W. Robust H-infinity control for uncertain stochastic systems with state delay. *IEEE Trans. Autom. Control* **2002**, *47*, 2089–2094.
21. Hua, M.; Zhang, F.; Deng, F.; Fei, J.; Chen, H. Mixed H₂/H_∞ control for discrete-time periodic Markov jump systems with quantization effects and packet loss compensation. *Nonlinear Anal. Hybrid Syst.* **2023**, *50*, 101398. [[CrossRef](#)]
22. Uchiyama, M. Formation of high-speed motion pattern of a mechanical arm by trial. *Trans. Soc. Instrum.* **1978**, *14*, 706–712.
23. Boudjedir, C.E.; Boukhetala, D.; Bouri, M. Iterative Learning Control of a Parallel Delta Robot. In *Advanced Control Engineering Methods in Electrical Engineering Systems*; Springer International Publishing: New York, NY, USA, 2018; pp. 72–83.
24. You, D.; Cui, L.; Bu, X.; Zhao, X.; Hou, R. Longitudinal Slippage Iterative Learning Trajectory Tracking Control of Wheeled Robots. *Electron. Optics Contr.* **2022**, *29*, 96–101.
25. Xi, W.; Wang, Y.; Chen, B.; Wu, H. Iterative learning control of robot based on artificial bee colony algorithm. *Proc. Ins. Mech. Eng. Part I J. Syst. Contr. Eng.* **2018**, *233*, 1221–1238. [[CrossRef](#)]
26. Liu, F.; Fan, Y. Higher-Order PID Sampled-Data Iterative Learning Control. *Contr. Eng. China* **2012**, *19*, 73–76.

27. Wang, H.; Dong, J.; Wang, Y. Research on Open-Closed-Loop Iterative Learning Control with Variable Forgetting Factor of Mobile Robots. *Discret. Dyn. Nat. Soc.* **2016**, *2016*, 6452179. [[CrossRef](#)]
28. Cao, W.; Dai, X.; Liu, Y. Discrete Iterative Learning Control with Variable Forgetting Factor for Mobile Robots. *J. Beijing Uni. Technol.* **2015**, *41*, 1516–1521.
29. Norouzi, A.; Koch, C.R. Robotic Manipulator Control Using PD-type Fuzzy Iterative Learning Control. In Proceedings of the 2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE), Edmonton, AB, Canada, 5–8 May 2019; pp. 1–4.
30. Patan, K.; Patan, M. Neural-network-based High-order Iterative Learning Control. In Proceedings of the 2019 American Control Conference, Philadelphia, PA, USA, 10–12 July 2019; pp. 2873–2878.
31. Wu, G.; Zhang, X.; Zhu, L.; Lin, Z.; Liu, J. Fuzzy Sliding Mode Variable Structure Control of a High-speed Parallel PnP Robot. *Mech. Mach. Theory* **2021**, *162*, 104349. [[CrossRef](#)]
32. Li, Q.; Wu, G.; Shen, H. Trajectory tracking control of SCARA parallel robot based on fuzzy adaptive iterative learning algorithm. In Proceedings of the Third International Conference on Mechanical Design and Simulation (MDS 2023), Xi'an, China, 3–5 March 2023; p. 126392N-(1–8).
33. Valentine, F.A. A Lipschitz condition preserving extension for a vector function. *Am. J. Math.* **1945**, *67*, 83–93. [[CrossRef](#)]
34. Li, Q.; Liu, E.; Cui, C.; Wu, G. An open-closed-loop iterative learning control for trajectory tracking of a high-speed 4-dof parallel robot. *Intell. Robot.* **2022**, *2*, 89–104. [[CrossRef](#)]
35. Zadeh, L. Fuzzy algorithms. *Inf. Control* **1968**, *12*, 94–102. [[CrossRef](#)]
36. Åström, K.J.; Wittenmark, B. *Adaptive Control*; Courier Corporation: Chelmsford, MA, USA, 2013.
37. Khalil, H.K. Lyapunov stability. *Control. Syst. Robot. Autom.* **2009**, *12*, 115.
38. Ando, T. Matrix young inequalities. *Oper. Theory Adv. Appl* **1995**, *75*, 33.
39. Wu, G.; Zhao, W.; Zhang, X. Optimum time-energy-jerk trajectory planning for serial robotic manipulators by reparameterized quintic NURBS curves. *Proc. Ins. Mech. Eng. Part C-J. Mech. Eng. Sci.* **2021**, *235*, 4382–4393. [[CrossRef](#)]
40. Gauthier, J.F.; Angeles, J.; Nokleby, S. Optimization of a test trajectory for SCARA systems. In *Advances in Robot Kinematics: Analysis and Design*; Springer: Dordrecht, The Netherlands, 2008; pp. 225–234.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.