IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 27, 2019, accepted May 22, 2019, date of publication May 30, 2019, date of current version June 13, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2920020

Force/Position Hybrid Control for a Hexa Robot Using Gradient Descent Iterative Learning Control Algorithm

BA-PHUC HUYNH, CHENG-WEI WU, AND YONG-LIN KUO
Graduate Institute of Automation and Control, National Taiwan University of Science and Technology, Taipei 10607, Taiwan
Corresponding author: Yong-Lin Kuo (kuo@mail.ntust.edu.tw)

This work was supported in part by the Ministry of Science and Technology, Taiwan, under Grant MOST 107-2221-E-011-135.

ABSTRACT This paper presents an approach of the force/position hybrid control for a hexa parallel robot to guarantee a safe and accurate interaction when touching the object surface. A double-loop PID controller is proposed to replace the common PID controller in the position control to eliminate position errors due to the dynamics coupling effect between the arms and the vibration of the mechanical system. An impedance control model is used to guarantee a safe and accurate interaction when touching the object surface. In addition, a gradient descent iterative learning control algorithm is used and modified to determine the optimal impedance parameters in unknown environments. A model of the robot is built in SimMechanics to simulate and estimate system parameters. After that, the experimental work was conducted on a real robot to verify the effectiveness and feasibility of the proposed method.

INDEX TERMS Hybrid control, Hexa robot, impedance control, double-loop PID, iterative learning.

I. INTRODUCTION

The parallel manipulators [1] have become popular since Stewart [2] proposed a new design for a flight simulation. Over the years, several structures have been studied and presented. Pierrot *et al.* [3] and Uchiyama [4] focused on a six-degree-of-freedom parallel structure, which is known by the name of the Hexa robot. Its design is a solution for pick-and-place operations in industry because of its high stiffness, high speed and acceleration, high accuracy and high carrying capability. However, it is complicated to calculate kinematics and dynamics [5], [6]. While the solution for the inverse kinematics problem can be found, the forward kinematics problem is almost impossible to solve analytically [7]. The reachable workspace of the Hexa robot also needs to be calculated based on the rotation limitations of the joints to avoid collisions of links [6] or detect singular positions, which may cause damage of the robot [8].

Many concepts, methods and algorithms have been proposed to control industrial robots and make them suitable for various types of work. The position control is offered in most commercial manipulators. The force control is used in situations where environmental interaction needs to be controlled. Impedance control is a type of indirect force control method and most frequently implemented in interaction control because of its robustness and feasibility [12]–[16]. However, impedance control affects position control and causes position errors. Rather than performing a decoupled control of both position and force variables, the hybrid approach describes a combination of position data and force/torque information to obtain desired control trajectory. The implementation of hybrid position/force control in robot arms [17] was studied to filling the gap between theory and industrial applications. Raibert and Craig [18] used analysis, simulation, and experiments to evaluate the ability of hybrid position/force control in executing trajectory of manipulators. A hybrid control strategy for dual-arm object manipulation using fused force/position errors and iterative learning [19] was proposed to grasp and move various objects. Shi *et al.* [30] used a hybrid controller to achieve the desired end-effector motion and contact forces required for driving a screw into a floating target. A force/position control structure of a five degree of freedom redundant actuation parallel robot using an improved fuzzy model predictive control algorithm is proposed by Wen *et al.* [31]. The authors [32] introduce position/torque hybrid control for a newly designed

The associate editor coordinating the review of this manuscript and approving it for publication was Dong Shen.

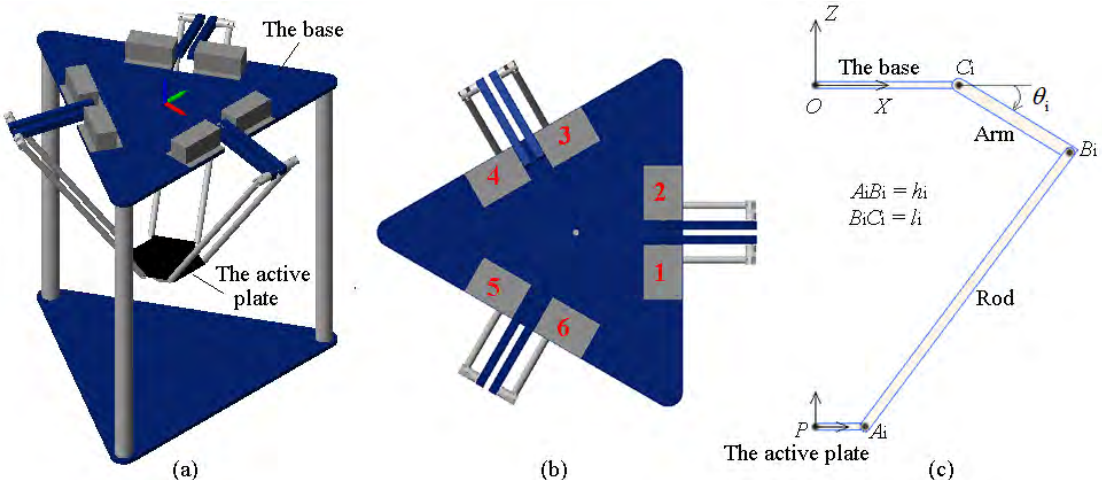

FIGURE 1. The model of Hexa parallel robot.

rigid and high-gear ratio quadruped robot. Another application of hybrid force/position control is on parallel machine for mechanical test [33]. However, there are not many studies about the application of hybrid control for Hexa robots. The complex structure of parallel robot causes a bit difficulty that the extra errors behind the position control loop due to the dynamics coupling effect between the arms of Hexa robot need to be handled. In this paper, a hybrid control architecture is proposed for the Hexa robot. The combination of position control and force control ensures a safe and accurate interaction of the robot when touching the object surface. To eliminate the extra errors due to the dynamics coupling effect between the arms of Hexa robot, a double-loop PID controller for the position control is developed in place of common PID controller. This improvement brings high efficiency in experiment, especially for robots with imperfect mechanical structures.

In impedance control method, the interaction force is changed from environment to environment and even within a same environment over time. This causes difficulty in finding desired parameters of the impedance controller. Therefore, some studies developed learning schemes, in which impedance parameters are regulated through a repetitive learning process. The natural actor-critic algorithm is used for impedance learning in robotic contact tasks [21], but just simulating in 2-D space. The authors derived a PI^2 algorithm for learning variable impedance control [22]. This algorithm requires calculations related to the velocity and torque of joints, which are very complex in a parallel manipulator such as Hexa robot. A new fuzzy-based impedance control strategy is developed by Tran *et al.* [23], but it needs a big data preprocessing for learning. The proposed method of Li and Ge [24] is a complex approach of impedance learning control. The authors tried to develop a learning law based on the sensory feedback of position and velocity errors instead of modeling the environment. The proposed iterative learning control of impedance control in this paper is developed from a gradient descent algorithm. The learning law is based on both position and force errors, therefore, it is compatible with the hybrid control methods used for the Hexa robot. The combination of position and force errors tracking in the learning law ensures that the position and interaction force will be adjusted gradually during the learning process. Due to the characteristic of the gradient descent algorithm, the cost function will decrease during the iterations, making the impedance parameters converge to the desired optimal value.

Based on the above discussion, this study proposes an implementation of controlling the Hexa robot with the primary contributions as follows:

- A force/position hybrid controller is developed for controlling the trajectory of the Hexa parallel robot, guaranteeing a safe and accurate interaction when the robot touches the object surface. The control approach is simulated in MATLAB SimMechanics and evaluated experimentally using a real Hexa robot system.
- An iterative learning control law is derived from the gradient descent algorithm to find optimal parameters of impedance control in unknown environments.
- A double-loop PID controller for position control is developed to replace a common PID controller to eliminate the position error due to the dynamics coupling effect between the arms and the vibration of mechanical system.

The following content of this paper includes: Section II describes the kinematics problems of the robot. Section III describes the hybrid controller and the iterative learning control algorithm. Section IV and V describe the setup of simulation and experiment, Section VI reports on the results, and Section VII presents conclusions.

II. KINEMATICS OF A HEXA ROBOT

Fig. 1 presents the model of the Hexa parallel robot. The design of Hexa parallel robot [3] has six equal kinematic

chains, which connect the base platform to the active plate. Each chain is described as a serial RUS chain, where R stands for revolute joint driven by a motor, U is for universal or cardan joint and S is for spherical joint in the active plate. There are two coordinate systems on the Hexa robot. The fixed frame $\{O\}$ is positioned in the center of the base platform with Z axis perpendicular to the base upwards. The X axis direction is pointed to the center point of C_1C_2 edge. The Y axis determined by the right-hand rule. The active frame $\{P\}$ is positioned in the center of the active plate with Z axis perpendicular to the base upwards. The X axis direction is pointed to the center point of A_1A_2 edge. The Y axis determined by the right-hand rule. For each kinematic chain*i*,for $i = 1 \ldots 6$, there are:

- The point A_i represents the center of the spherical joint that connect the rod to the active plate.
- The point B_i represents the centers of the universal joints that connect the rods to the arms.
- The point C_i represents the rotation center of revolute joint of the arm.
- The angle θ_i is the rotation angle of revolute joint of arm.
- The lengths of arm and rod are l_i and h_i, respectively.

The forward and inverse kinematics equations must be determined with the aim of allowing the active plate position and rotation angles of arms can be freely converted. Inverse kinematics uses the position and orientation vector of the active plate to derive the rotation angle of each arm. Forward kinematics determines the position and orientation of the active plate based on the rotation angles of arms.

A. INVERSE KINEMATICS

In inverse kinematics, the coordinate of the center point P (x_p, y_p, z_p) and the Euler angles (α, β, γ) of the active plate with respect to the fixed frame $\{O\}$ are given. The coordinate on the active frame $\{P\}$ can be transformed to the coordinate on the fixed frame $\{O\}$ by using the rotation matrix R:

$$R = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} \tag{1}$$

where

$$\begin{aligned} R_{11} &= \cos\alpha \cdot \cos\beta \\ R_{12} &= \cos\alpha \cdot \sin\beta \cdot \sin\gamma - \sin\alpha \cdot \cos\gamma \\ R_{13} &= \cos\alpha \cdot \sin\beta \cdot \cos\gamma + \sin\alpha \cdot \sin\gamma \\ R_{21} &= \sin\alpha \cdot \cos\beta \\ R_{22} &= \sin\alpha \cdot \sin\beta \cdot \sin\gamma + \cos\alpha \cdot \cos\gamma \\ R_{23} &= \sin\alpha \cdot \sin\beta \cdot \cos\gamma - \cos\alpha \cdot \sin\gamma \\ R_{31} &= -\sin\beta \\ R_{32} &= \cos\beta \cdot \sin\gamma \\ R_{33} &= \cos\beta \cdot \cos\gamma \end{aligned}$$

The coordinate of point A_i with respect to the frame $\{O\}$ can be calculated as follow:

$$A_i^O = P^O + R \cdot A_i^P \tag{2}$$

where A_i^O is the coordinate of the spherical joint i with respect to the frame $\{O\}$; P^O is the coordinate of the center point of the active plate with respect to the frame $\{O\}$; A_i^P is the coordinate of the spherical joint i with respect to the frame $\{P\}$.

The point C_i is fixed on the base platform. The relationship between B_i and C_i is as follow:

$$\begin{bmatrix} x_{B_i} \\ y_{B_i} \\ z_{B_i} \end{bmatrix} = \begin{bmatrix} x_{C_i} \\ y_{C_i} \\ z_{C_i} \end{bmatrix} + \begin{bmatrix} l_i \cos\lambda_i \cos\theta_i \\ -l_i \sin\lambda_i \cos\theta_i \\ -l_i \sin\theta_i \end{bmatrix} \tag{3}$$

where

$$\begin{aligned} &[\,\lambda_1 \;\; \lambda_2 \;\; \lambda_3 \;\; \lambda_4 \;\; \lambda_5 \;\; \lambda_6\,] \\ &= \begin{bmatrix} 0 & 0 & \frac{2\pi}{3} & \frac{2\pi}{3} & \frac{4\pi}{3} & \frac{4\pi}{3} \end{bmatrix} \end{aligned} \tag{4}$$

The relationship between A_i and B_i can be expressed as:

$$h_i^2 = (x_{B_i} - x_{A_i})^2 + (y_{B_i} - y_{A_i})^2 + (z_{B_i} - z_{A_i})^2 \tag{5}$$

Substituting equations (1) to (4) into equation (5), a new equation can be derived in the format:

$$a = b \cdot \cos\theta + c \cdot \sin\theta \tag{6}$$

So that, the rotation angle θ_i of the arm can be calculated as follows:

$$\theta_{\mathrm{i}} = \arctan\left[\frac{a_i c_i^2 - b_i\sqrt{-c_i^2(a_i^2 - b_i^2 - c_i^2)}}{c_i[a_i b_i + \sqrt{-c_i^2(a_i^2 - b_i^2 - c_i^2)}]}\right] \tag{7}$$

with

$$\begin{cases} a_i = \dfrac{h_i^2 - l_i^2 - x_{CA_i}^2 - y_{CA_i}^2 - z_{CA_i}^2}{2l_i} \\ b_i = x_{CA_i} \\ c_i = -z_{CA_i} \\ i = 1, 2 \end{cases} \tag{8}$$

$$\begin{cases} a_i = \dfrac{h_i^2 - l_i^2 - x_{CA_i}^2 - y_{CA_i}^2 - z_{CA_i}^2}{l_i} \\ b_i = \sqrt{3}y_{CA_i} - x_{CA_i} \\ c_i = -2z_{CA_i} \\ i = 3, 4 \end{cases} \tag{9}$$

$$\begin{cases} a_i = \dfrac{h_i^2 - l_i^2 - x_{CA_i}^2 - y_{CA_i}^2 - z_{CA_i}^2}{l_i} \\ b_i = -\sqrt{3}y_{CA_i} - x_{CA_i} \\ c_i = -2z_{CA_i} \\ i = 5, 6 \end{cases} \tag{10}$$

where

$$\begin{aligned} x_{CA_i} &= x_{C_i} - x_{A_i} \\ y_{CA_i} &= y_{C_i} - y_{A_i} \\ z_{CA_i} &= z_{C_i} - z_{A_i} \end{aligned}$$

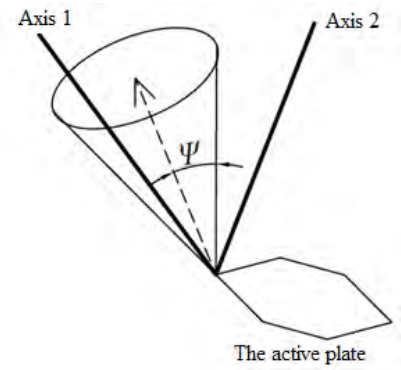

FIGURE 2. The spherical joint angle limit.

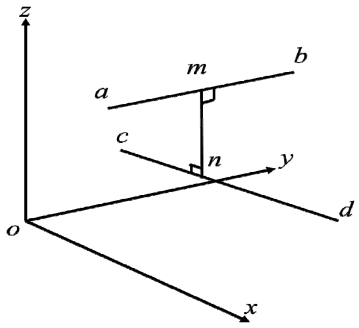

FIGURE 3. The distance between two lines.

1) WORKSPACE

The workspace is the reachable positions of the active plate. Depending on the requirements, there are some types of workspace. In this study, three conditions are considered [6]:

a: THE MATHEMATICAL LIMIT

Based on trigonometry, equation (6) must satisfy the condition:

$$\left|\frac{a}{\sqrt{b^2+c^2}}\right| \leq 1 \tag{11}$$

b: ANGULAR JOINTS LIMITS

The angle limit between the rod and the arm connected to the spherical joint in the active plate is shown in Fig. 2. The spherical joint angles are calculated as:

$$\{l_i\}.\{h_i\} = \|\{l_i\}\| \,.\, \|\{h_i\}\| \,.\, \cos\phi_i \tag{12}$$

thus:

$$\phi_i = \arccos\left(\frac{\{l_i\}.\{h_i\}}{\|l_i\| \,.\, \|h_i\|}\right) \tag{13}$$

where φ_i is the angle between rod and arm.

The dashed arrow pointing to the outside of the active plate indicates a normal vector, thereby calculating the maximum allowed angle of the spherical joints. Ψ defines the boundary (cone format) in which the rods can be positioned with respect to the active plate. Axis 1 means that the rod is in the allowed angle, so it is still in the working space, but Axis 2 means that the rod is not in the allowed range, so it violates the physical principle and is therefore not working.

c: AVOIDING COLLISION

After applying the conditions (*a*) and (*b*), almost all types of collision are eliminated. However, the collision between rods may occur. When analyzed using numerical method, the rods like line segments. The minimum distance between them must be calculated and compared to a minimum safe distance value. If two rods are parallel, the distance is calculated like a distance between a point and a line. If they are not, the distance is calculated as two lines, such as shown in Fig. 3 and equation (14).

$$|\vec{mn}| = \frac{\left|\vec{bd}.\vec{ab} \times \vec{cd}\right|}{\left|\vec{ab} \times \vec{cd}\right|} \tag{14}$$

2) SINGULARITY

When the parallel mechanical arm is at a singular point, the force on the joint will increase sharply, resulting in insufficient torque of the motor and deformation of the joint and the connecting rod. Therefore, it is necessary to perform singular point analysis and to avoid the singular points of the arms. This paper uses the power inspired measure method [8] to detect and safely control the Hexa robot out of the singularities.

B. FORWARD KINEMATICS

In the forward kinematics problem, the rotation angles of motors are given. The orientation and Euler angles of the active plate are needed to be determined to satisfy equation (7). The Hexa robot is a system of coupled nonlinear equations. In general, finding a unique closed form analytic solution to such a system is very complicated. Therefore, in this study, the artificial neural network (ANN) approach [7] is proposed. Fig. 4 shows the procedure of ANN training.

First, a data set of positions of the active plate is created randomly. This data set must be modified to reject the singular points and points outside the workspace as discussed in Section II.A. The modified data set is called the target data in ANN training.

From the target data, using equations (7) to (10), the data set of angles θ_i of six arms is created. This data set is called the input data in ANN training. The neural network training tool (*nntool*) in *MATLAB* is used for back propagation training. The training results in weight and bias factors which can be used to calculate FKP in the hybrid control process.

III. HYBRID CONTROL AND ITERATIVE LEARNING

Force/position hybrid control is the combination of position and force information into one control scheme to control the end effector in unknown environment. From the original control scheme of Craig and Raibert [20] and the correction

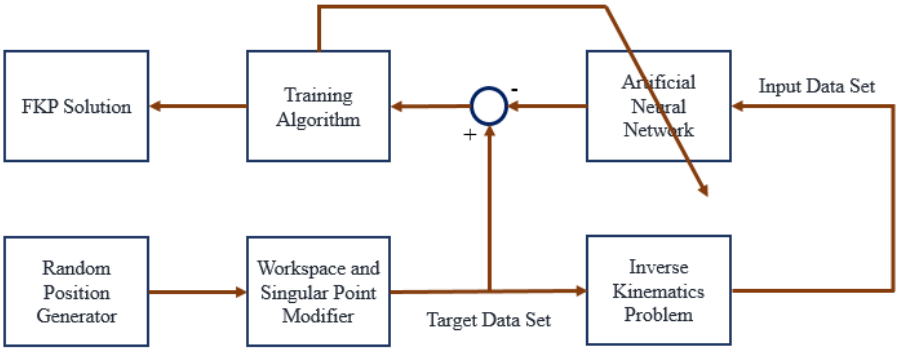

FIGURE 4. The procedure of the neural network training.

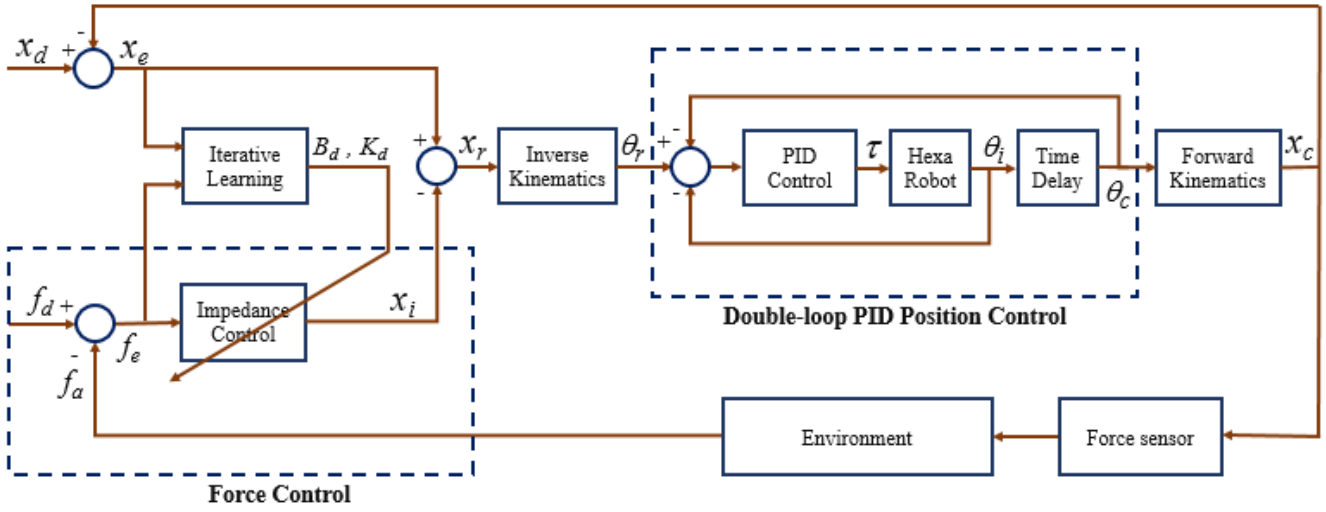

FIGURE 5. The control architecture.

of Fisher and Mujtaba [29], this study develops an architecture of hybrid control using an impedance controller for force control and an improved PID controller for position control, which is shown in Fig. 5. The x_i, x_c, x_d, x_e and x_r are 6×1 vectors including both position and orientation information. the f_a, f_d and f_e are 6×1 vectors including force and moment information. The θ_r, θ_i and θ_c are 6×1 vectors representing the rotation angles of six arms as presented in Section II.A. The τ is 6×1 vector representing torques applied to the arms. The B_d and K_d are parameters of the impedance control, which participate indirectly in the control process.

The desired positions of arms are calculated as:

$$\theta_r = T \cdot x_r \tag{15}$$

where T is transformation matrix of inversed kinematics problem. x_r is reference position of the end effector, which is calculated by the position error x_e of the end effector and the position compensator x_i of the impedance control.

The torques applied to the arms can be expressed as:

$$\tau = \Gamma(\theta_r) \tag{16}$$

where Γ is position compensator transfer function. θ_r is reference position.

The current position of the end effector is calculated as:

$$x_c = J.\theta_c \tag{17}$$

where θ_c is the current rotation angles of arms. J is the manipulator Jacobian matrix. However, in this study, J is replaced by calculation based on the weight matrices and bias vector, which are determined by the ANN training of forward kinematics problem presented in Section II.B.

The position error of the end effector is calculated as:

$$x_e = x_d - x_c \tag{18}$$

where x_d is the desired position of the end effector.

The relationship between the current position of the end effector and the contact force can be expressed as:

$$f_a = \Lambda(x_c) \tag{19}$$

where Λ is a transfer function. In *SimMechanics*, it is the function inside force sensor block that simulates the contact force. In experiment, it is the function inside the real device defined by the manufacturer. In both cases, the output value of the force sensor is considered rather than the function Λ.

The equations (15) and (16) show that the drive signals of the arms are based on both force and position information. Meanwhile, equations (17), (18) and (19) present instantaneous effect of controlling the arms on the satisfaction of both force and position constraints. This is the basic idea of the hybrid controller. The detail of each component in the hybrid controller will be presented in next subsections.

A. POSITION CONTROL

The position control block in Fig. 5 consists of DC motors and drivers. Based on Newton's 2nd law and Kirchhoff's voltage law, the mathematical relationship in the electric equivalent circuit of DC motor can be derived as:

$$V = Ri + L\frac{di}{dt} + K_b\dot{\theta} \tag{20}$$

$$J\ddot{\theta} + B\dot{\theta} = K_t i \tag{21}$$

where V is armature voltage, i is armature current, R is electric resistance, L is electric inductance, e is the back electromotive force, B is motor viscous friction constant, J is moment of inertia of the rotor, T is the torque generated by a DC motor, K_t is motor torque constant, K_b is electromotive force constant, ω is angular velocity, θ is the rotational angle of motor.

Applying the Laplace transform into equations (20) and (21), the open-loop transfer function of motor can be expressed as:

$$G(s) = \frac{\theta(s)}{V(s)} = \frac{K_t}{s[(Js+B)(Ls+R)+K_bK_t]} \tag{22}$$

Because the motor electric inductance L is generally extremely small compared to R, and therefore can be ignored, equation (22) can be rewritten as:

$$G(s) = \frac{\theta(s)}{V(s)} = \frac{K_t}{s[R(Js+B)+K_bK_t]} = \frac{K_t}{RJs^2+(RB+K_bK_t)s} \tag{23}$$

To use system identification, the transfer function in equation (23) is expressed in the following format:

$$G(s) = \frac{\theta(s)}{V(s)} = \frac{c}{s^2+as+b} \tag{24}$$

where a, b, and c are model parameters, which will be determined by using the *MATLAB System Identification Toolbox*. The setup of system identification will be discussed in Section IV.A and the result will be shown in Section VI.

A PID controller is used in the feedback control system as shown in Fig. 6. The PID parameters K_p, K_i and K_d are determined by using the Ziegler–Nichols tuning method in *MATLAB Simulink*.

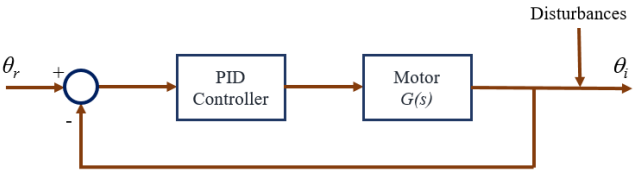

FIGURE 6. Block diagram of common PID position control.

B. DOUBLE-LOOP PID CONTROL USING TIME DELAY

In the experimental work, after the common PID control was performed and the motor reached the desired position, the control loop is stopped. After that, because of disturbances, the position of the motor is slid a bit and this causes the position error. The angle is changed from θ_i to θ_c. The disturbances can be formed by the following factors:

- The dynamics have not been considered in the control system, so the inertial and gravitational effects must affect the positioning accuracy.
- The backlash of motor gearbox and the vibration of the mechanical structure also affect the accuracy.
- The dynamics coupling effect between the arms causes a change of position.

To solve this problem, a double-loop control is proposed as shown in Fig. 7. After performing the first feedback loop, the controller will wait for a certain time to make sure the position error occurs. Then it will call the second feedback loop to adjust the position. The double-loop position control applies well to robots with imperfect mechanical systems. Its effectiveness will be shown in Section VI.

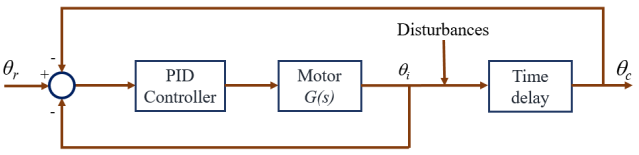

FIGURE 7. Block diagram of double-loop PID position control.

C. IMPEDANCE CONTROL

The model of impedance control can be given as:

$$M_d(\ddot{x}_d - \ddot{x}_c) + B_d(\dot{x}_d - \dot{x}_c) + K_d(x_d - x_c) = f_e \tag{25}$$

where the desired inertia, damping, and stiffness parameter matrices are represented by M_d, B_d, and K_d, respectively; x_d is the desired trajectory of the plate; x_c is the current trajectory of the plate in the position control loop; $f_e = f_d - f_a$ is the force error; f_d is the desired interaction force of environment; f_a is the current interaction force of environment in the position control loop.

To determine the dynamic behavior of the system, the damping ratio ζ is calculated as follow:

$$\zeta = \frac{B_d}{2\sqrt{K_dM_d}} \tag{26}$$

The damping ratio is very important for contact stability. For application in this paper, it is required a significantly overdamped impedance behavior to ensure a stable contact with a stiff environment. Therefore, the parameters M_d, B_d and K_d are selected so that the damping ratio $\zeta > 1$. In iterative learning control process, the parameter $\zeta$is selected at a certain interval depending on the real system, while the parameters M_d, B_d and K_d will be optimized with the condition of satisfying equation (26).

The objective of hybrid control in this study is to guarantee a safe and accurate interaction between the robot and object surface that considered as the environment. Therefore, the value of f_d is set as close to zero as possible. In the experiment, the interaction force of environment f_a is measured by the force/torque sensor. In the simulation, a model of the environment in *SimMechanics* will produce the interaction force f_a.

D. ITERATIVE LEARNING CONTROL

To determine the optimal parameters of the impedance control for unknown environments, this paper proposes a gradient descent iterative learning control (ILC) law, which can change the impedance parameters through repetitive interactions to improve the performance of the impedance control, so guarantee a safe and accurate interaction of the robot when touching the object surface.

In the learning process, the parameter M_d is fixed by the apparent inertia of manipulator. The CAD of the robot can estimate approximately the inertia value. Our simulation and experiment process shows that the arbitrary change of M causes instability and the impedance parameters will be difficult to converge to the desired optimal value.

The parameters B_d and K_d are modified and updated through the iterative learning process. The gradient descent ILC algorithm [25] is used to develop the learning law. The general form of this algorithm is expressed as follow:

$$u^{k+1} = u^k + \beta \left(G^k\right)^T e^k \tag{27}$$

where k is the repetition number; u is the input applied to the learning process; β is the learning gain; G is transfer function of nominal model; e is the output error; The product of $G^T e$ determines the direction of the update vector.

The objective of impedance control block is force tracking, so the output error is calculated as:

$$e^k = f_e^k - f_e^{k-1} \tag{28}$$

The parameters B_d and K_d are the inputs in equation (27). The transfer functions G are calculated based on the gradient scheme to ensure gradual change of the interaction force by updating B_d and K_d as follow:

$$\begin{aligned} G_B^k &= \frac{\partial f_e^k}{\partial B_d^k} \\ G_K^k &= \frac{\partial f_e^k}{\partial K_d^k} \end{aligned} \tag{29}$$

Substituting f_e in equation (25) into equation (29), the formulations of B_d and K_d can be obtained as follow:

$$\begin{aligned} G_B^k &= \left(\dot{x}_d^k - \dot{x}_c^k\right)^T = \left(\dot{x}_e^k\right)^T \\ G_K^k &= \left(x_d^k - x_c^k\right)^T = \left(x_e^k\right)^T \end{aligned} \tag{30}$$

Applying equations (27), (28) and (30) into our system, the learning law is formed as follow:

$$\begin{aligned} B_d^{k+1} &= B_d^k + \beta_B \dot{x}_e^k \left(f_e^k - f_e^{k-1}\right) \\ K_d^{k+1} &= K_d^k + \beta_K x_e^k \left(f_e^k - f_e^{k-1}\right) \end{aligned} \tag{31}$$

Convergence is a concerned matter when applying ILC algorithms. [25], [34] analyze the convergence of unknown nonlinear systems in detail. Applying to this paper, the convergence of equation (27) is guaranteed if:

$$\left|1 - \beta \left(G^k\right)^T\right| < 1 \tag{32}$$

Applying to equation (31), the convergence conditions are:

$$\begin{aligned} \left|1 - \beta_B \dot{x}_e^k\right| &< 1 \\ \left|1 - \beta_K x_e^k\right| &< 1 \end{aligned} \tag{33}$$

In general, the position error changes after each iteration and different from zero, so the conditions in equation (33) can be satisfied. In practical, the equation (33) will be checked in each iteration and the learning gains are adjusted if necessary to guarantee the convergence.

IV. THE SETUP OF SIMULATION

The simulation study includes two parts: the identification of system parameters for the position control and the proposed impedance iterative learning control.

A. THE IDENTIFICATION OF SYSTEM PARAMETERS FOR THE POSITION CONTROL

To implement the system identification for the motors as discussed in Section III.A, the input signals are generated by feeding the motor a sine sweep. The output signals are the rotating angles of motors that can be measured by the encoders and recorded in the SD card. The system identification is completed using the *MATLAB System Identification Toolbox*. Fig. 8 is the block diagram of the system identification.

The PID parameters K_p, K_i and K_d are determined by using the Ziegler–Nichols tuning method in *Simulink*.

B. THE IMPEDANCE CONTROL AND ITERATIVE LEARNING CONTROL

The model of Hexa robot is built in *SimMechanics* (*MATLAB*) with the same size and parameters to the real robot. This model is used to simulate the process of the proposed impedance iterative learning control in Section III.D. Fig. 9 shows the block diagram of the model.

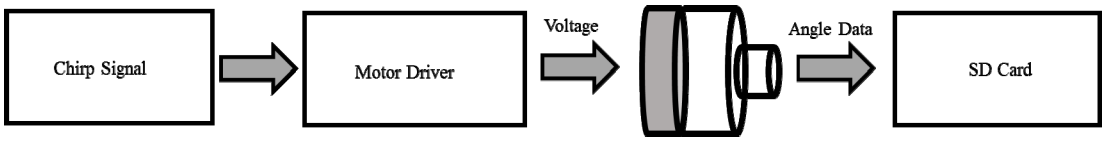

FIGURE 8. Block diagram of system identification.

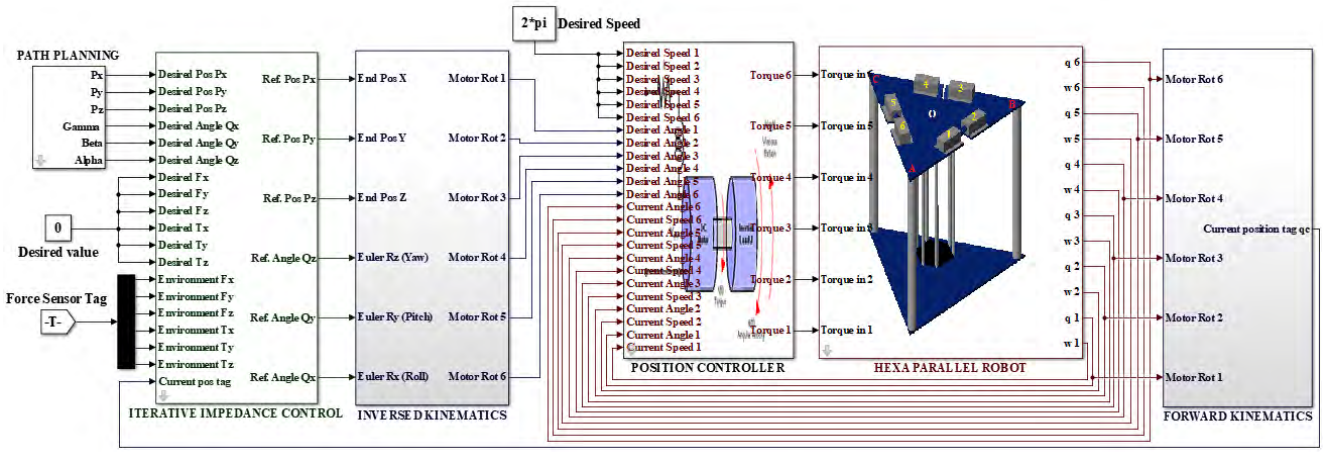

FIGURE 9. The *SimMechanics* model of the Hexa parallel robot.

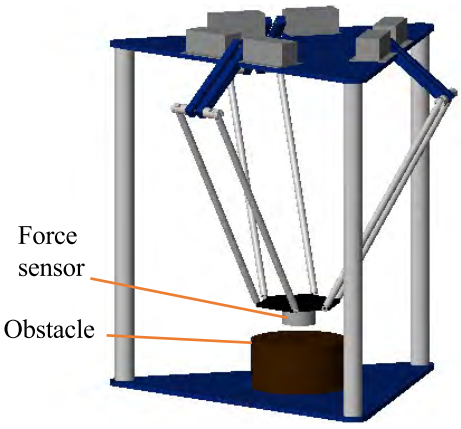

FIGURE 10. The impedance iterative learning control process in *SimMechanics*.

Fig. 10 is the simulation of the impedance iterative learning control in *SimMechanics*. The robot structure, force sensor and obstacle are blocks inside the Hexa parallel robot block that is shown in Fig. 9. The active plate is controlled to move and touch the obstacle. The new values of B_d and K_d will be updated based on the interaction force and the position error as in equation (31). The learning process is performed forty iterations. The parameters of position control are the estimated values in Section IV.A. M_d is fixed and equals to the apparent inertia. The initial value of the damping and stiffness parameters are $B_d^1 = 1$ and $K_d^1 = 1$. The learning gains are $\beta_B = 0.001$ and $\beta_K = 0.0005$. In the learning process, the values of B_d, K_d and interaction force are logged for evaluation.

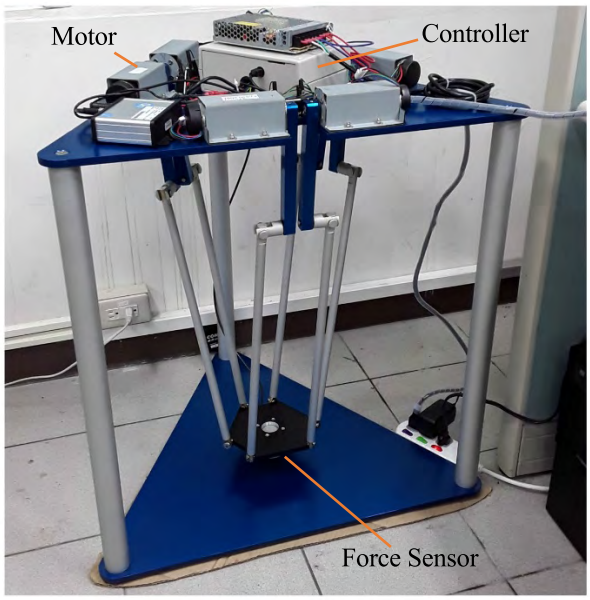

FIGURE 11. The Hexa parallel robot.

V. THE SETUP OF EXPERIMENT

Fig. 11 is the real Hexa parallel robot used in the experiment. Fig. 12 is the block diagram of the control system. The PC and control software are to control the robot via RS232. The software is written in *Lazarus*. Fig. 13 shows the interface of the software. The robot controller is an *Arminno* development board of Holtek that includes a Cortex M3 ARM embedded system. The board communicates with the software via

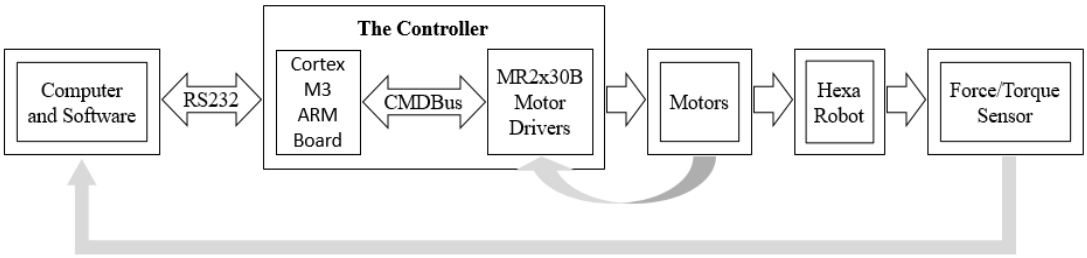

FIGURE 12. The block diagram of the control system.

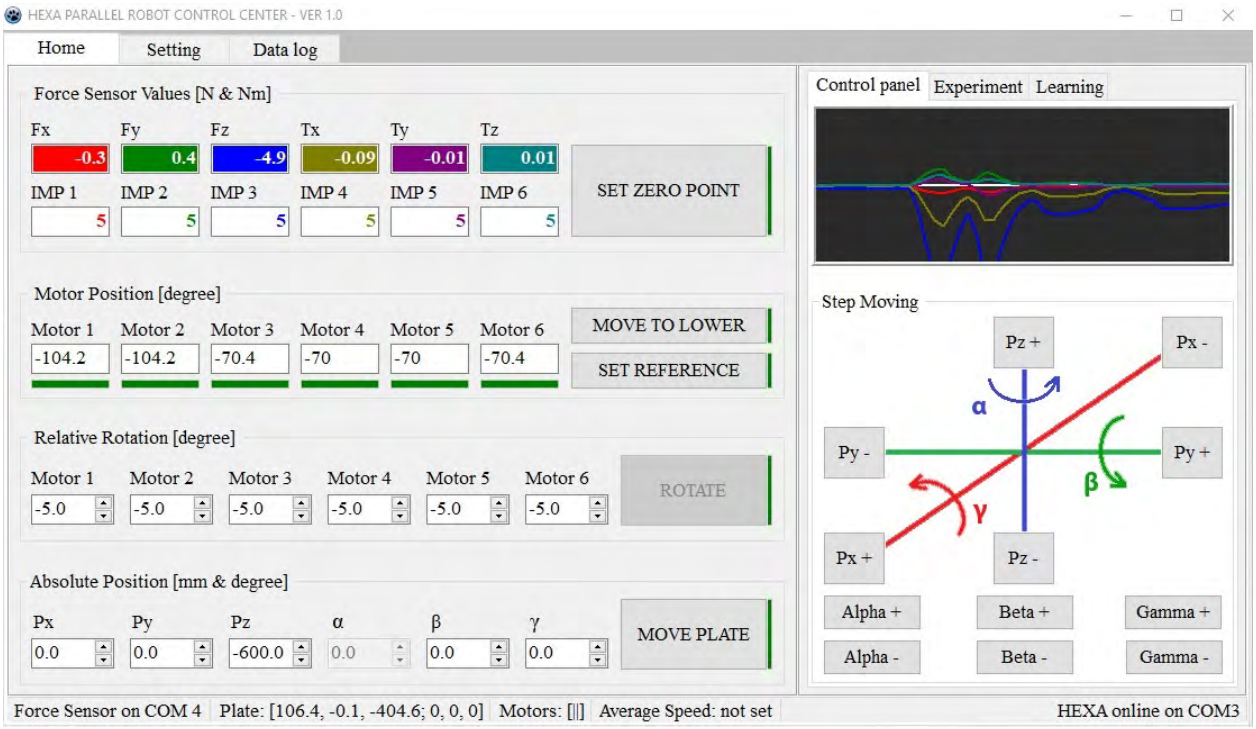

FIGURE 13. The control software of the Hexa parallel robot.

RS232 and controls the motors of the robot. The motors are IG-42GM models, manufactured by Sha Yang Ye, Inc, Taiwan. The motor includes a gear box with the reduction ratio of 1/49. The rated voltage, current, torque, and speed of the motor are 12 V, 5500 mA, 18 kgcm, and 119 r/min, respectively. The motor is attached by an incremental encoder, which uses two Hall-effector sensors to measure the rotating angle of the motor. The accuracy of the encoder is 0.367°. The motor controller includes MR2x30B motor driving boards. The force sensor is a HEX-E 6-Axis Force/Torque sensor of OnRobot.

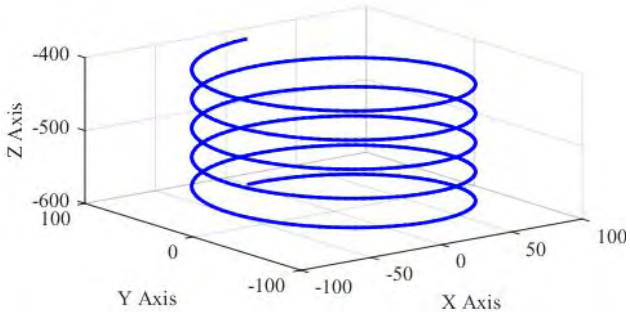

FIGURE 14. Helix trajectory.

A. THE POSITION CONTROL

In this experiment, the robot is controlled to move the active plate a helix trajectory in the work space, which is shown in Fig. 14. To evaluate the effectiveness of the double-loop PID control discussed in Section III.B, the position control is implemented in two cases. In the first case, a common PID controller is performed. In the second case, the double-loop PID controller is used. The positions of motors are tracked continuously in both cases. Finally, the real tracked

trajectories will be compared together to evaluate the performance of the position controller.

B. THE IMPEDANCE CONTROL AND ITERATIVE LEARNING

Similar to the simulation work, the impedance iterative learning process is performed to find the optimal value of impedance control parameters. The robot is controlled to move the plate to touch an obstacle. The interaction force and position error will be read to update the parameters in equation (31). The learning process is performed in forty iterations. In the experimental work, the force is read from a real force sensor. M_d is also fixed and equals to the apparent inertia. At the first iteration ($k = 1$), the damping and stiffness parameters are $B_d^1 = 5$ and $K_d^1 = 5$. The learning gains are $\beta_B = 0.001$ and $\beta_K = 0.0005$. The values of B_d, K_d and interaction force are logged for evaluation.

FIGURE 15. Moving on the sloping surface in the hybrid control experiment.

C. HYBRID CONTROL

To evaluate the effectiveness of the proposed hybrid control approach with the estimated system parameters and the impedance parameters obtained in the iterative learning process, the Hexa robot is controlled to move the plate along a sloping surface as shown in Fig. 15. The exact position and slope of the surface is not known. The control program must hold plate always lightly touch the surface with a small interactive force. Force tracking data will be saved for analysis.

VI. DISCUSSION OF SIMULATION AND EXPERIMENTAL RESULTS

Table 1 shows the estimated parameters of six motors of the Hexa robot by using the system identification and Table 2 shows the PID parameters by using the Ziegler–Nichols tuning method in *Simulink* as discussed in Section IV.A. These parameters will be used for other implementations of the robot.

Fig. 16 to Fig. 21 show the comparisons of the position control for using the double-loop PID control with using a common PID control when the Hexa robot moves the

TABLE 1. The parameters of system model of motors.

	Motor 1	Motor 2	Motor 3	Motor 4	Motor 5	Motor 6
a	34.92	33.94	38.34	34.65	34.79	34.87
b	0.07	0.06	0.12	0.07	0.17	0.01
c	37.69	35.85	40.29	35.14	36.97	36.93

TABLE 2. The PID controller parameters.

	Motor 1	Motor 2	Motor 3	Motor 4	Motor 5	Motor 6
Kp	14.5345	14.5275	14.5127	14.5216	14.5268	14.5331
Ki	0.6045	0.6032	0.6063	0.6055	0.6048	0.6046
Kd	0.5012	0.5024	0.5019	0.5032	0.5021	0.5017

helix trajectory that is setup in Section V.A. As discussed in Section III.B, disturbances occur behind the PID control loop and thus generate position errors. In the case of using the double-loop PID control, when the plate's position deflects from the desired path, the double-loop control adjusts to bring it back to the correct position. The adjustment process is carried out continuously to keep the path as expected. In fact, the position deviation takes place very fast with a very small range and is adjusted immediately so it is almost impossible to see. Table 3 shows the comparisons of resulted errors for the double-loop PID control with the normal PID control. In the case of using the double-loop PID control, *mse*, *mae* and *nrmse* errors are much lower than the case of using the common PID control.

Fig. 22 to Fig. 24 show the results of the impedance iterative learning in both simulation and experiment. Note that this is the tracked data of moving in Z axis. The data tracking is similar for the remaining axes. The learning process is performed in forty iterations. Depending on the model of the robot, the number of iterations and the learning gains can be changed. Table 4 summarizes the parameters of the learning process.

In the simulation, the robot moves the trajectory in thirteen seconds. From 0 to 1.8 seconds, the force is zero and the parameters B_d and K_d are constant because the plate has not touched the obstacle. From 1.8 seconds, the interaction force increases as the plate touches the obstacle. The learning process tries to update the impedance parameters to reduce the interaction force. At the end of the trajectory, the values of B_d and K_d are recorded and used as the initial value for next iteration. As in Fig. 22, in the first iteration, the interaction force is big, about 15 N. However, after forty iterations, the interaction force becomes smaller while the impedance parameters are changed. At the end of learning process, the parameters are $B_d^{40} = 0.3826$ and $K_d^{40} = 0.0001$, the interaction force is 1.4303 N. The interaction force is small and acceptable when it ensures the robot only touches the object lightly.

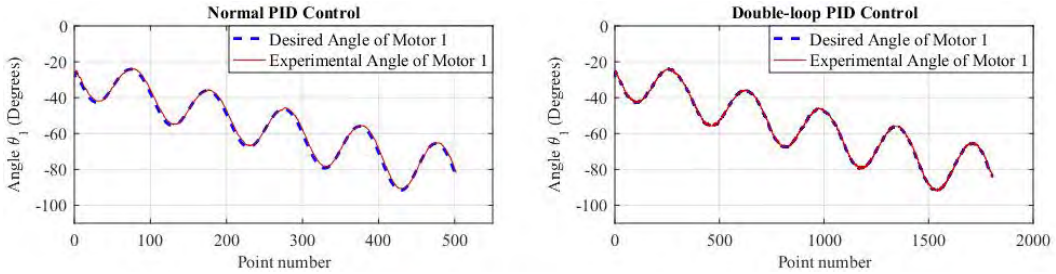

FIGURE 16. Comparison of motor angle θ_1 for the double-loop PID control (right) with the normal PID control.

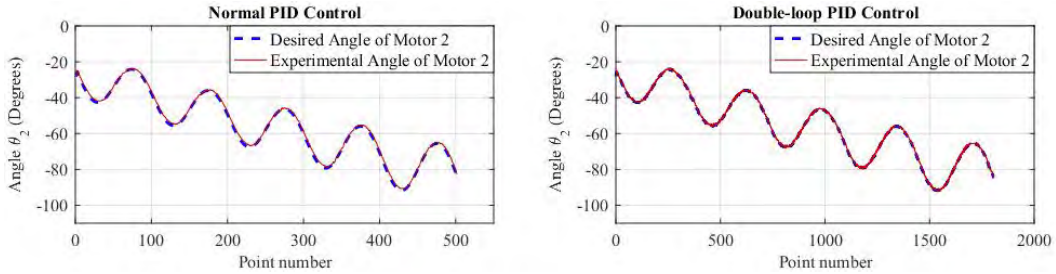

FIGURE 17. Comparison of motor angle θ_2 for the double-loop PID control (right) with the normal PID control.

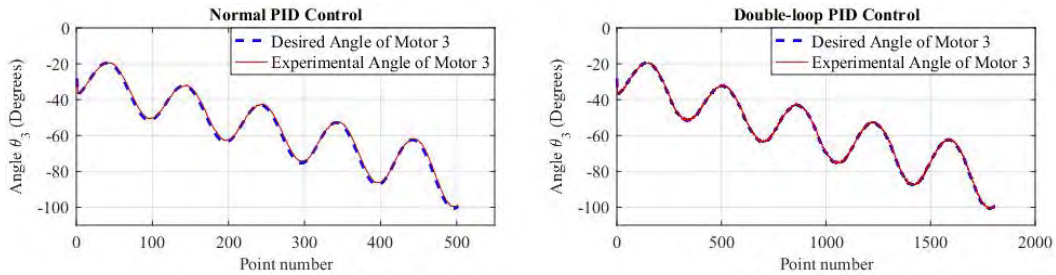

FIGURE 18. Comparison of motor angle θ_3 for the double-loop PID control (right) with the normal PID control.

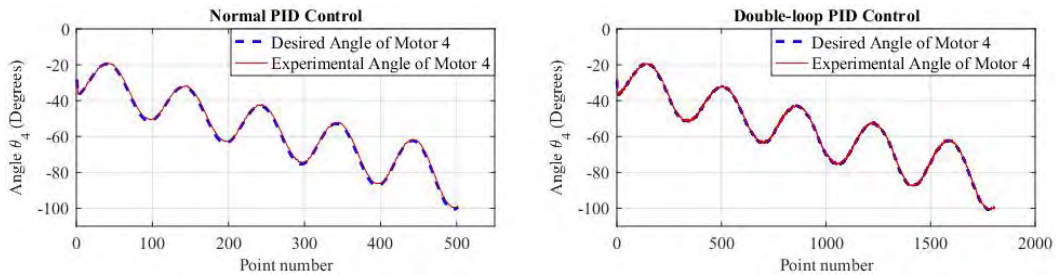

FIGURE 19. Comparison of motor angle θ_4 for the double-loop PID control (right) with the normal PID control.

Similar to the simulation, the experimental result shows that the impedance parameters are gradually improved during the iterative learning to reduce the interaction force. At the end of learning process, the parameters are $B_d^{40} = 0.5386$ and $K_d^{40} = 0.0005$, the interaction force is kept about 1.6256 N. Note that, the chart of force is not smooth because of the vibration of the robot and the noise of the force sensor. This causes the oscillation of the parameters B_d and K_d in the learning process. However, the parameters still converge to the desired value.

At the first iteration, the contact force is quite large. The reason is that the initial values of B_d and K_d were selected far from optimal values. To reduce contact force in the first iteration, the initial values of B_d and K_d should be estimated and chosen as close to the optimal value as possible. A good estimation is based on the optimal value of parameters B_d

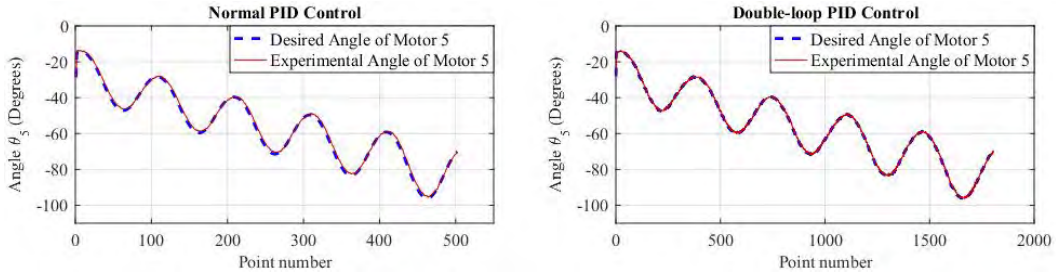

FIGURE 20. Comparison of motor angle θ_5 for the double-loop PID control (right) with the normal PID control.

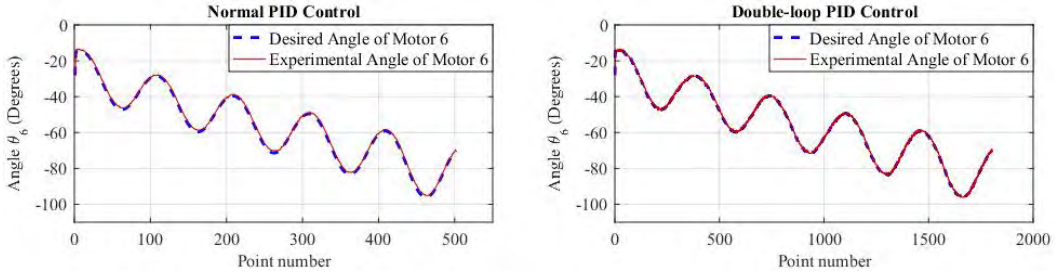

FIGURE 21. Comparison of motor angle θ_6 for the double-loop PID control (right) with the normal PID control.

TABLE 3. Comparison of resulted errors for the double-loop PID control with the normal PID control.

θ	*MSE*		*MAE*		*NRMSE*	
	Normal PID	Double-loop PID	Normal PID	Double-loop PID	Normal PID	Double-loop PID
1	1.64230	0.24507	1.00289	0.33689	0.01917	0.00728
2	0.95313	0.22764	0.82670	0.33918	0.01460	0.00698
3	1.39438	0.23438	0.95088	0.33019	0.01468	0.00594
4	0.93740	0.22605	0.82503	0.33611	0.01203	0.00583
5	1.21030	0.23187	0.90147	0.33084	0.01343	0.00585
6	0.85714	0.22733	0.82351	0.34077	0.01135	0.00577

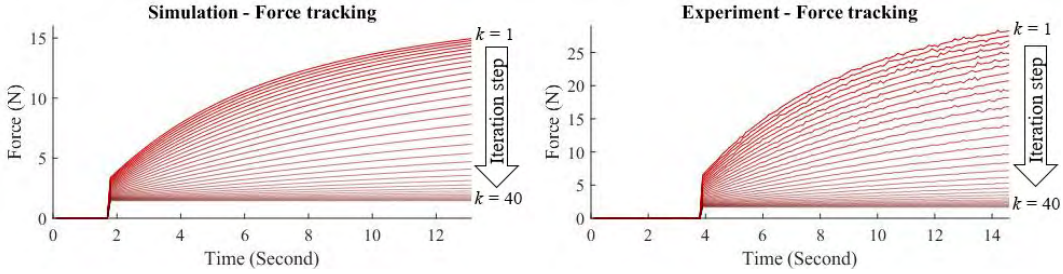

FIGURE 22. Interaction force of the iterative learning process in simulation (left) and experiment (right).

and K_d found in the simulation to set the initial value for the experiment.

The change of learning gains will also change the convergence speed. In practice, these learning gains are adjusted depending on the specific system to achieve optimal results. Another note is that the learning gains may be adjusted in each iteration to guarantee the convergence as presented in section III.D.

Fig. 25 shows the result of the hybrid control experiment as discussed in Section V.C. The maximum force and torque

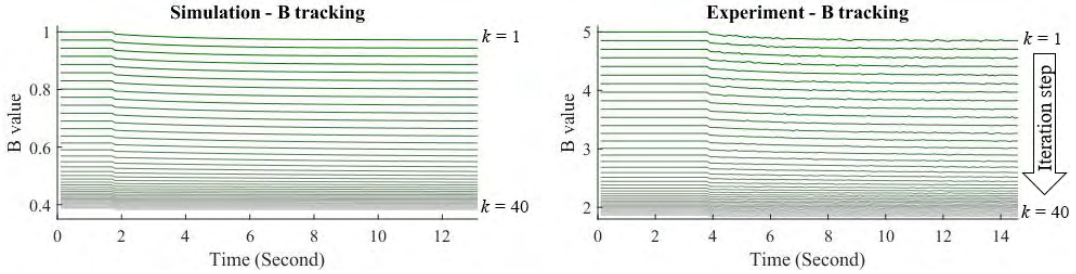

FIGURE 23. Damping parameter of the iterative learning process in simulation (left) and experiment (right).

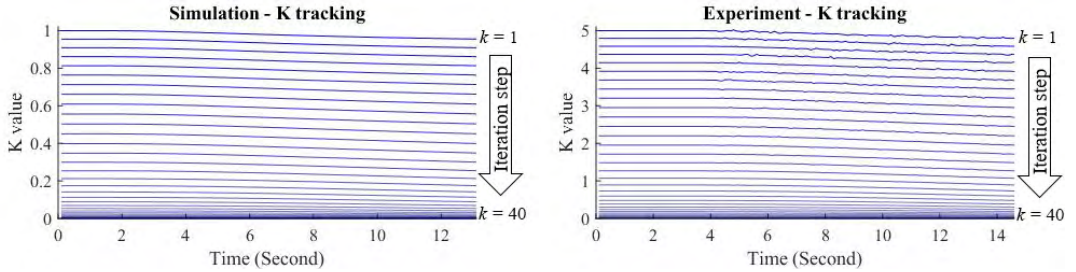

FIGURE 24. Stiffness parameter of the iterative learning process in simulation (left) and experiment (right).

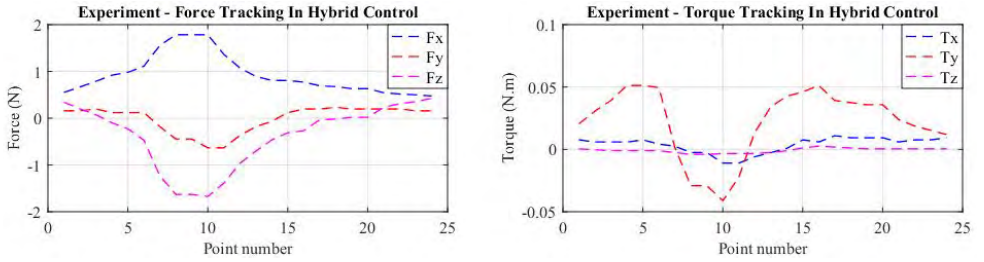

FIGURE 25. The force tracking in the hybrid control experiment.

TABLE 4. The summary of parameters in the learning process.

	Simulation	Experiment
Number of iterations	40	40
Learning gain β_B	0.001	0.001
Learning gain β_K	0.0005	0.0005
Initial Bd	1	5
Initial Kd	1	5
Final Bd	0.3826	0.5386
Final Kd	0.0001	0.0005
Final Interaction force (N)	1.4303	1.6256

are 1.785 N of F_x and 0.0512 Nm of T_y, respectively. These values mean that the plate only touches lightly on the surface of the obstacle.

VII. CONCLUSIONS

In this study, a force/position hybrid control is proposed for controlling trajectory of the Hexa parallel robot and guaranteeing a safe and accurate interaction when the robot touches the object surface. The double-loop PID controller is developed to replace a common PID controller to eliminate the position error due to the dynamics coupling effect between the arms and the vibration of mechanical system. The iterative learning control law is derived from the gradient descent algorithm to find the optimal values of the impedance parameters in unknown environments. The proposed approach has been implemented in both simulation and experiment to verify its validity. The results show that this approach can control the robot as desired.

REFERENCES

[1] J. P. Merlet, *Parallel Robots*, 2nd ed. Dordrecht, The Netherlands: Springer, 2006.
[2] D. Stewart, "A platform with six degrees of freedom," *Proc. Inst. Mech. Eng.*, vol. 180, no. 1, pp. 371–386, 1965.
[3] F. Pierrot, P. Dauchez, and A. Fournier, "HEXA: A fast six-DOF fully-parallel robot," in *Proc. 5th Int. Conf. Adv. Robot.*, Pisa, Italy, Jun. 1991, pp. 1158–1163. doi: 10.1109/ICAR.1991.240399.
[4] M. Uchiyama, "A 6 d.o.f. parallel robot HEXA," *Adv. Robot.*, vol. 8, no. 6, p. 601, Jan. 1993.
[5] X. Wang and Y. Tian, "Inverse dynamics of Hexa Parallel Robot based on the Lagrangian equations of first type," in *Proc. Int. Conf. Mechanic Automat. Control Eng.*, Wuhan, China, Jun. 2010, pp. 3712–3716. doi: 10.1109/MACE.2010.5535969.
[6] S. Celso, F. Tartari, E. Lobo, and E. Cabral, "Kinematics and workspace analysis of a parallel architecture robot: The Hexa," in *Proc. Int. Congr. Mech. Eng.*, Ouro Preto, Brasil, vol. 2, Nov. 2005, pp. 158–165.

[7] M. Dehghani, M. Ahmadi, A. Khayatian, M. Eghtesad, and M. Farid, "Neural network solution for forward kinematics problem of Hexa parallel robot," in *Proc. Amer. Control Conf.*, Seattle, WA, USA, Jun. 2008, pp. 4214–4219. doi: 10.1109/ACC.2008.4587155.
[8] E. M. De Queiroz, C. C. Bier, A. Campos, J. Maaß, and R. Guenther, "Direct singularity avoidance strategy for the Hexa parallel robot," in *Proc. Int. Congr. Mech. Eng.*, Ouro Preto, Brasil, vol. 2, Nov. 2005, pp. 182–189.
[9] El H. Hasnaa and B. Mohammed, "Planning a trajectory of a 6-DOF parallel robot ≪HEXA≫," in *Proc. Int. Conf. Electr. Inf. Technol.*, Tangiers, Morocco, vol. 2, May. 2016, pp. 300–305. doi: 10.1109/EITech.2016.7519610.
[10] C. Vaida, D. Pisla, F. Covaciu, B. Gherman, A. Pisla, and N. Plitea, "Development of a control system for a HEXA parallel robot," in Proc. IEEE Int. Conf. Automat., Qual. Test. Robot., Cluj-Napoca, Romania, May 2016, pp. 1–6. doi: 10.1109/AQTR.2016.7501318.
[11] P. Last, C. Budde, C. Bier, and J. Hesselbach, "HEXA-parallel-structure calibration by means of angular passive joint sensors," in *Proc. IEEE Int. Conf. Mechatronic Automat.*, Niagara Falls, ON, Canada, Jul./Aug. 2005, pp. 1300–1305. doi: 10.1109/ICMA.2005.1626741.
[12] R. Ikeura and H. Inooka, "Variable impedance control of a robot for cooperation with a human," in *Proc. IEEE Int. Conf. Robot. Automat.*, Nagoya, Japan, May 1995, pp. 3097–3102. doi: 10.1109/ROBOT.1995.525725.
[13] J. De Gea and F. Kirchner, "Modelling and simulation of robot arm interaction forces using impedance control," *IFAC Proc. Volumes*, Seoul, South Korea, vol. 41, no. 2, pp. 15589–15594, Jul. 2008. doi: 10.3182/20080706-5-KR-1001.02636.
[14] F. Caccavale, P. Chiacchio, A. Marino, and L. Villani, "Six-DOF impedance control of dual-arm cooperative manipulators," *IEEE/ASME Trans. Mechatronics*, vol. 13, no. 5, pp. 576–586, Oct. 2008.
[15] Z. Li, J. Liu, Z. Huang, Y. Peng, H. Pu, and L. Ding, "Adaptive impedance control of human–robot cooperation using reinforcement learning," *IEEE Trans. Ind. Electron.*, vol. 64, no. 10, pp. 8013–8022, Oct. 2017. doi: 10.1109/TIE.2017.2694391.
[16] G. Raiola, C. A. Cardenas, T. S. Tadele, T. De Vries, and S. Stramigioli, "Development of a safety- and energy-aware impedance controller for collaborative robots," *IEEE Robot. Autom. Lett.*, vol. 3, no. 2, pp. 1237–1244, Apr. 2018. doi: 10.1109/LRA.2018.2795639.
[17] G. Ferretti, G. Magnani, and P. Rocco, "Toward the implementation of hybrid position/force control in industrial robots," *IEEE Trans. Robot. Autom.*, vol. 13, no. 6, pp. 838–845, Dec. 1997. doi: 10.1109/70.650162.
[18] M. H. Raibert and J. J. Craig, "Hybrid position/force control of manipulators," *J. Dyn. Syst., Meas., Control*, vol. 103, no. 2, pp. 126–133, 1981. doi: 10.1115/1.3139652.
[19] B.-H. Chen, Y.-H. Wang, and P.-C. Lin, "A hybrid control strategy for dual-arm object manipulation using fused force/position errors and iterative learning," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatron.*, Auckland, New Zealand, Jul. 2018, pp. 39–44. doi: 10.1109/AIM.2018.8452248.
[20] J. J. Craig and M. H. Raibert, "A systematic method of hybrid position/force control of a manipulator," in *Proc. Comput. Softw. IEEE Comput. Soc.'s 3rd Int. Appl. Conf.*, Chicago, IL, USA, Nov. 1979, pp. 446–451. doi: 10.1109/CMPSAC.1979.762539.
[21] B. Kim, J. Park, S. Park, and S. Kang, "Impedance learning for robotic contact tasks using natural actor-critic algorithm," *IEEE Trans. Syst., Man, Cybern. B Cybern.*, vol. 40, no. 2, pp. 433–443, Aug. 2009. doi: 10.1109/TSMCB.2009.2026289.
[22] J. Buchli, F. Stulp, E. A. Theodorou, and S. Schaal, "Learning variable impedance control," *Int. J. Robot. Res.*, vol. 30, pp. 820–833, Apr. 2011. doi: 10.1177/0278364911402527.
[23] H. T. Tran, H. Cheng, H. Rui, X. Lin, M. K. Duong, and Q. Chen, "Evaluation of a fuzzy-based impedance control strategy on a powered lower exoskeleton," *Int. J. Social Robot.*, vol. 8, no. 1, pp. 103–123, Jan. 2015.
[24] Y. Li and S. S. Ge, "Impedance learning for robots interacting with unknown environments," *IEEE Trans. Control Syst. Technol.*, vol. 22, no. 4, pp. 1422–1432, Jul. 2013. doi: 10.1109/TCST.2013.2286194.
[25] C. T. Freeman, E. Rogers, J. H. Burridge, A.-M. Hughes, and K. L. Meadmore, "Iterative learning control—An overview," in *Iterative Learning Control for Electrical Stimulation and Stroke Rehabilitation*. London, U.K.: Springer, 2015, pp. 3–15.
[26] P. K. Jamwal, S. Hussain, M. H. Ghayesh, and S. V. Rogozina, "Impedance control of an intrinsically compliant parallel ankle rehabilitation robot," *IEEE Trans. Ind. Electron.*, vol. 63, no. 6, pp. 3638–3647, Jun. 2016. doi: 10.1109/TIE.2016.2521600.
[27] X. Lu, Y. Zhao, and M. Liu, "Self-learning interval type-2 fuzzy neural network controllers for trajectory control of a Delta parallel robot," *Neurocomputing*, vol. 283, pp. 107–119, Mar. 2018. doi: 10.1016/j.neucom.2017.12.043.
[28] D. H. Owens, J. Hatonen, and S. Daley, "Robust gradient-based iterative learning control," in *Proc. IEEE Int. Conf. Netw., Sens. Control*, London, U.K., Apr. 2007, pp. 163–168. doi: 10.1109/ICNSC.2007.372770.
[29] W. D. Fisher and M. S. Mujtaba, "Hybrid position/force control: A correct formulation," *Int. J. Robot. Res.*, vol. 11, no. 4, pp. 299–311, Aug. 1992.
[30] L. Shi, J. Katupitiya, and N. M. Kinkaid, "Hybrid control of space robot in on-orbit screw-driving operation," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 54, no. 3, pp. 1253–1264, Jun. 2018. doi: 10.1109/TAES.2017.2780558.
[31] S. Wen, J. Chen, G. Qin, Q. Zhu, and H. Wang, "An improved fuzzy model predictive control algorithm based on the force/position control structure of the five-degree of freedom redundant actuation parallel robot," *Int. J. Adv. Robotic Syst.*, vol. 15, no. 5, pp. 1–8, Oct. 2018. doi: 10.1177/1729881418804979.
[32] O. Sim, T. Jung, K. K. Lee, J. Oh, and J.-H. Oh, "Position/torque hybrid control of a rigid, high-gear ratio quadruped robot," *Adv. Robot.*, vol. 32, no. 18, pp. 969–983, Sep. 2018. doi: 10.1080/01691864.2018.1516162.
[33] J. L. Flohic, F. Paccot, N. Bouton, and H. Chanal, "Application of hybrid force/position control on parallel machine for mechanical test," *Mechatronics*, vol. 49, pp. 168–176, Feb. 2018. doi: 10.1016/j.mechatronics.2017.12.006.
[34] K. L. Moore, "Iterative learning control: An expository overview," in *Applied and Computational Control, Signals, and Circuits*, vol. 1. New York, NY, USA: Springer, 1999, pp. 151–214.

BA-PHUC HUYNH received the B.E. degree in mechanical engineering from the Ho Chi Minh University of Technology, Vietnam, in 2003, and the M.S. degree in precision mechatronics engineering from the Minghsin University of Science and Technology, Hsinchu, Taiwan, in 2010. He is currently pursuing the Ph.D. degree in automation and control with the National Taiwan University of Science and Technology, Taipei, Taiwan.

From 2003 to 2008, he was a Lecturer with Can Tho University, Can Tho, Vietnam. From 2011 to 2015, he was the Technical Manager of Big C Hypermarket, Can Tho. From 2015 to 2017, he was the Deputy Technical Director of Pham Huynh Kien Giang Co., Ltd., Kien Giang, Vietnam. In 2017, he joined Kien Giang University, Kien Giang, where he is currently a Lecturer. His research interests include industrial robots, microcontrollers programming and applications, mobile robots, intelligent robots, and machine vision.

CHENG-WEI WU received the B.S. degree in automatic control engineering from Feng Chia University, Taiwan, in 2016, and the M.S. degree in automation and control from the National Taiwan University of Science and Technology, Taiwan, in 2018. His research interests include robot control, impedance control, force control, system identification, and dynamic modeling.

YONG-LIN KUO received the Ph.D. degree in mechanical engineering from the University of Toronto, Canada, in 2005. He is currently an Associate Professor with the National Taiwan University of Science and Technology, Taiwan. His research interests include system dynamics and control, robot manipulators, spacecraft dynamics and control, computational solid mechanics, kineto-elasto-dynamics, and enhanced finite element method.