The Study of Reduction in the Tracking Error of Robotic Arms using Iterative Learning Control

Manutsawee Kiewongart*, Manas Sangwotasil, Phichitphon Chotikunnan, Nuntachai Thongpance and Rawiphon Chotikunnan

College of Biomedical Engineering, Rangsit University, Pathum Thani, Thailand
*Corresponding author, E-mail: Leawming2008@hotmail.com

Abstract

This paper presents the ILC control system in the form of P-type, D-type, and PD-type applications and compare the decrease in RMSE of each type of ILC by focusing on the simulation results of the robot system in a defined motion pattern for the dive a real robot in the future. In the experimental design, the movement patterns of the robot were defined by motors A, B, and C. The results of the robot movement are taken to the transfer function for use in the simulation. An example of this research has shown the use of the techniques of P-type, D-type, and PD-type ILC. It was found that the system simulation results of P-type and PD-type ILC had similar reductions in RMSE values. However, PD-type ILC is adapted to reduce RMSE within the system, which is faster than P-type and achieves the lowest RMSE by just a little when compared to this simulation.

Keywords: *Iterative learning control, Robotic arms, Simulation*

1. Introduction

Today, industrial robots are important in the world. They can be applied to working in industries, hospitals, clinics, and many other places. Developing a robot that can move with minimal error is a problem that can be solved by implementing various theoretical approaches. The robot is assigned a movement to be directed by tracking a trajectory. The system always has the same error range when operated such as moving a batch of workpieces from one point to another in the batch process. When checking the root mean square error (RMSE) value, it is always apparent that there will be errors close to the original value.

Robotic control systems can be developed inside or outside the feedback control. For the development of the system inside the feedback control, the system can be flexible when a variety of profiles are modified, and it is an easy way to develop a controller.

To design a controller to control the joint or rotation of the motor, the research included proportional integral derivative (PID) (Pan et al., 2018), fuzzy logic (Cupertino et al., 2006; Liu et al., 2008), neural networks (Dexu et al., 2019), LQR (Sun et al., 2010; Van et al., 2011; Mason et al., 2014), sliding mode (Ferrara et al., 2019) and etc., to develop the movement of the motor resulting in a fast response, flexibility, and more robustness of the system. Considering the issue of RMSE arising from the movement could not solve all problems. The inside feedback control is interesting to be developed first when the system designer is the developer of the robot.

Developing the desired path by the outside feedback control is another option because the user cannot change the internal control structure. The control system that can be developed uses iterative learning control (ILC), repetitive control (RC), and run-to-run control. Iterative learning control is an interesting control system used in batch processes since it can fix the problem by decreasing the error in the next iteration (Bristow et al., 2006; Ahn et al., 2007). They are categorized into controls that can effectively handle tracking errors. By the basic concept, it can improve the ability of an inside feedback control by reducing the tracking errors in an iteration. There are two types of modulated control input signals: 1) controlled input signal outside the feedback control system in ILC serial architecture, and 2) controlled input signal inside the feedback control system in ILC parallel architecture. There are many types of ILC control systems, such as P-Type ILC, D-Type ILC (Wang, 2000; Song, et al., 2005), PD-Type ILC (Chen & Moore, 2002), and PID-Type ILC (Madady, 2008). With these different controller styles, developers often design learning matrix to suit the applications in the designed P, I, and D terms. There may be an approach for determining the error value as an option which can be designed and used to develop a system to work in the same way as developing a learning matrix. In addition, the selection of different types of ILC in the system has variations that are not the same as the responses within the system. In the design of the control system, there is no comparison among the performances of P-Type ILC, D-Type ILC, and PD-Type ILC in applications with robotic arms, in addition to the theoretical comparison of some controls only.

In this research, the difference of error to input in D-type ILC was designed by comparing the error values between iterations from the normal design D-type ILC by using it in different time steps as a whole before the improvement in PD-type. Then, P-Type ILC, D-Type ILC, and PD-Type ILC were compared in the aspect of their reduction of the RMSE value within the system. In terms of the scope of the study, one profile of motion was defined in 3 joints, and the design of the simulation system in the initial study showed the results of the following studies.

2. Objectives

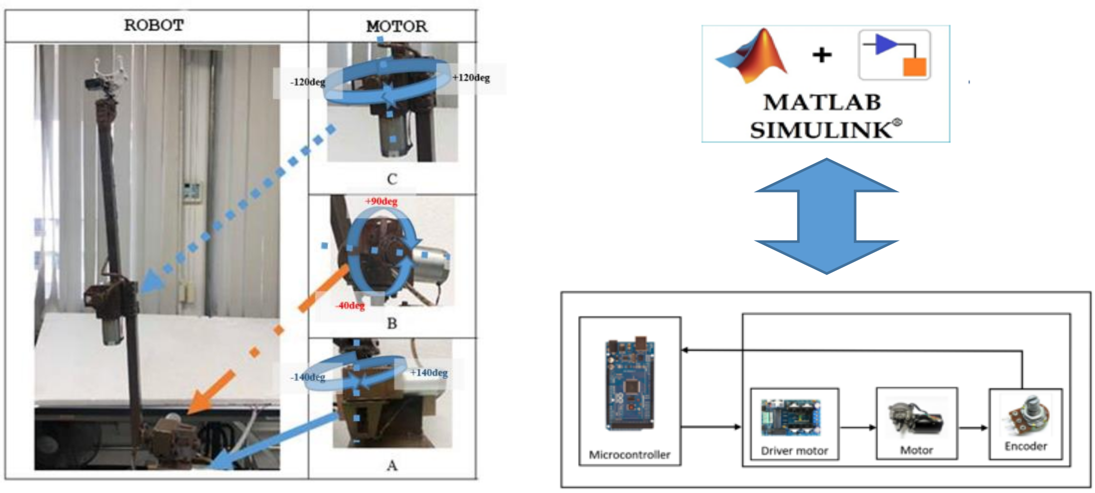
1) To study the use of the ILC controller in P-type, D-type, and PD-type

2) To compare the reduction of the error value of the ILC controller in the movement of the robot

3. Materials and Methods

3.1 Prototype robotic arm

According to Figure 1(a), the prototype robotic arm was used in this research. The robotic arm consisted of three axes of movement: Joint A, Joint B, and Joint C. Joint A was the pivot point of the robotic arm. Joint B was the lift up - lift down of the robotic arm, and Joint C was the sweep angle of the robot. The control processing system which was processed by the MATLAB/Simulink program and Arduino board received feedback by a variable resistor to check the angle of motion in each joint as shown in Figure 1(b).



(a) Prototype robotic arm (b) The system control process of the robotic

Figure 1 Prototype system robotic arm

3.2 Program design by MATLAB/Simulink

In the design of the control system, the researcher used P control because of the technique of Chien-Hrones–Reswick (CHR) tuning to select a Kp gain. The program in Simulink was employed to design the system into three parts for controlling Joint A, Joint B, and Joint C. By providing a method of control as a trajectory control as shown in Figure 2.

By setup, trajectory control can be defined through the variables in_m1, in_m2, and in_m3 after that program will run the process to control the robotics arm. For controller is responsible for processing in discrete PID control system and setting the limit of movement of each robotic arm as shown in the saturation box. the output of the robotics arm can get by the workspace box.

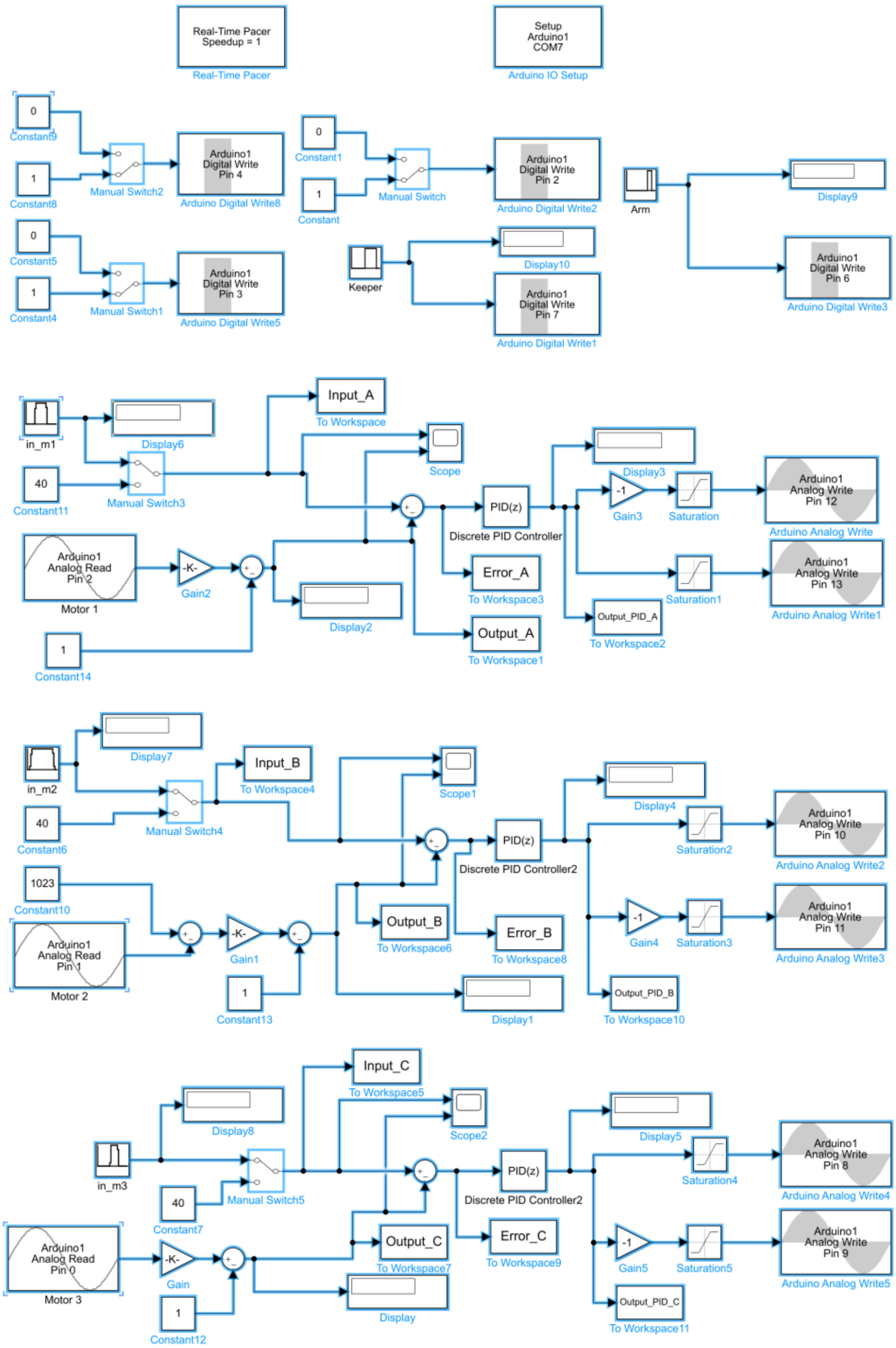
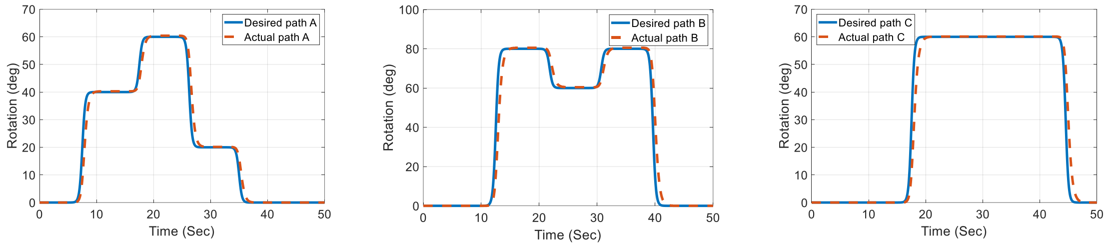


Figure 2 The control algorithm of the robotic arm using Simulink block

3.3 System equations of a prototype robotic arm

The system equations can find in the motion of robots, the experiment robot was moved by feedback control considering the desired path and actual path from the system run profile of movement in the time domain as shown in Figure 3.



(a) joint of motor A (b) joint of motor B (c) joint of motor C

Figure 3 Smooth the function profile to control the robotic arms in motor A, motor B, and motor C.

Form Figure 3 can estimate the transfer function of the robotics arm. By analyzing input and output from Figure 4 and using system identification from MATLAB Toolbox. The desired movement versus the actual movement of all joints from 49.94 sec from Figure 3(a) in joint of motor A, Figure 3(b) in joint of motor B, and Figure 3(c) in joint of motor C. Simulink program set up the time delay in the step input to 0.055 sec/step, and the batch process in trajectory control uses 909 steps in the path. The results of estimating the transfer function of a discrete-time system are defined as the ratio of the z transform of each joint according to joint of motor A is equations (1) in $\hat{G}_A(z)$, joint of motor B is equations (2) in $\hat{G}_B(z)$, and joint of motor C is equations (3) in $\hat{G}_C(z)$.

$$\hat{G}_A(z) = \frac{0.072767z}{z^2-1.350753z+0.423020} \quad (1)$$

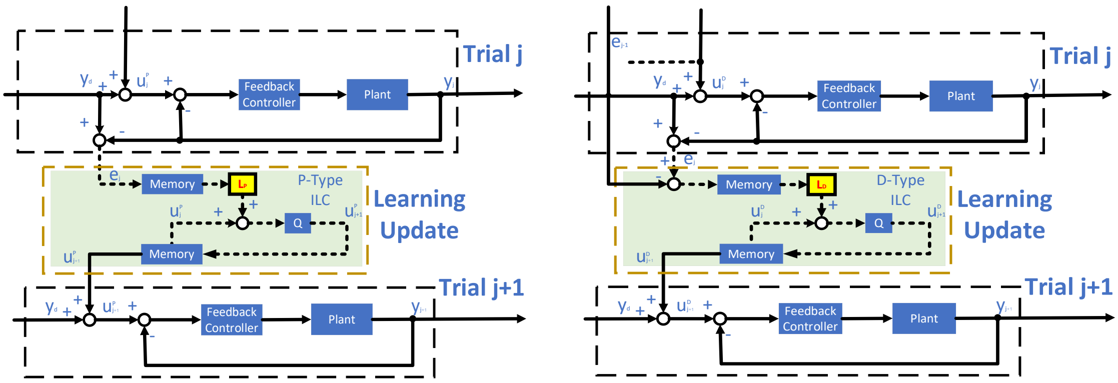
$$\hat{G}_B(z) = \frac{0.065985z}{z^2-1.350992z+0.416566} \quad (2)$$

$$\hat{G}_C(z) = \frac{0.059974z}{z^2-1.360473z+0.420346} \quad (3)$$

3.4 Iterative learning control

Iterative learning control is a control system based on a repeating pattern of movement motion. To update the control input, ILC uses the error value after iteration of the system to learn and update the data to estimate as a new control input for the next iteration.

Thus, the error value is reduced as the duty cycle increases. However, for this type of control the initial position of the system must always be set before feeding it as a control input to the system again in the next run in which the working principle of relearning control. As shown in Figure 6, the model of using this iterative learning control system is usually used in 2 forms: P-Type ILC and D-Type ILC.



(a) P-Type ILC (b) D-Type ILC

Figure 4 ILC serial architecture

The feedback control can be rewritten in the state space representation as

$$x_j(k+1) = Ax_j(k) + Bu_j(k)$$
$$y_j(k) = Cx_j(k) + Du_j(k)$$

where $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, and\ \boldsymbol{D}$ are the Markov parameters of the system, defining that $\boldsymbol{D} = 0$ for simplicity. The subscript j denotes the iteration number j. k is time step input. $\boldsymbol{u}_j$ is the input sequence, and the output sequence is $\boldsymbol{y}_j$.

In these results, $\boldsymbol{y}_j = \boldsymbol{P}\boldsymbol{u}_j$ where the matrix $\boldsymbol{P}$ represents the feedback control system.

$$\underbrace{\left[y_j(1) \ \vdots \ y_j(N)\right]}_{y_j} = \underbrace{\left[p_1 \ \dots \ 0 \ \vdots \ \ddots \ \vdots \ p_N \ \dots \ p_1\right]}_{P} \underbrace{\left[u_j(0) \ \vdots \ u_j(N-1)\right]}_{u_j} + \left[q_1 \ \vdots \right. \quad (4)$$

where p_i=$CA^{i-1}B$ and q_i=CA^i, for $i \in [1, N]$. N and $x_j(0)$ denote the initial sta simplicity.

From Figure 4 (a), a simple iterative learning control equation can be written in the form of control P-Type ILC as shown in equation (5);

$$u_j^P(k) = u_{j-1}^P(k) + \boldsymbol{L}_P e(k+1) \quad (5)$$

where $\boldsymbol{u}_j^P$ is control input of P-Type ILC and $\boldsymbol{L}_P$ is the learning control matrix of it the subscript k denotes the time step k.

In Figure 4 (b), there is also a D-Type ILC control with the equation shown in equation (6)

$$u_j^D(k) = u_{j-1}^D(k) + \boldsymbol{L}_D(e_j(k+1) - e_{j-1}(k+1)) \quad (6)$$

where $\boldsymbol{u}_j^D$ is control input of D-Type ILC and L_P is the learning control matrix of it

By the asymptotic convergence value can be found by using equation (7) and (8)

$$|\rho(I - PL)| < 1 \quad (7)$$
$$max\ \sigma(\boldsymbol{I} - \boldsymbol{PL}) < 1 \quad (8)$$

where (7) is an eigenvalue condition by ρ is spectral radius (maximum absolu eigenvalues of (***I-PL***) and (8) is the singular value condition by σ is maximum and minimu of matrix (***I-PL***). ***I*** is the identify matrix and ***L*** is the learning control matrix of iterative learning.

For this research, to test PD-Type ILC shown in Figure 5 and design a new D-Type ILC, the original errors of the difference in time steps in $\dot{e} = e_j(k+1) - e_j(k)$ were changed to approximate the errors of iteration differences in $\dot{e} = e_j(k+1) - e_{j-1}(k+1)$. The functional integration between P-Type ILC in basic usage and D-Type ILC is shown as equation (9).

$$u_j(k) = y_d(k) + (u_j^P(k) + u_j^D(k)) \quad (9)$$

where $\boldsymbol{u}_j$ is control input, $\boldsymbol{y}_d$ is the desired path.

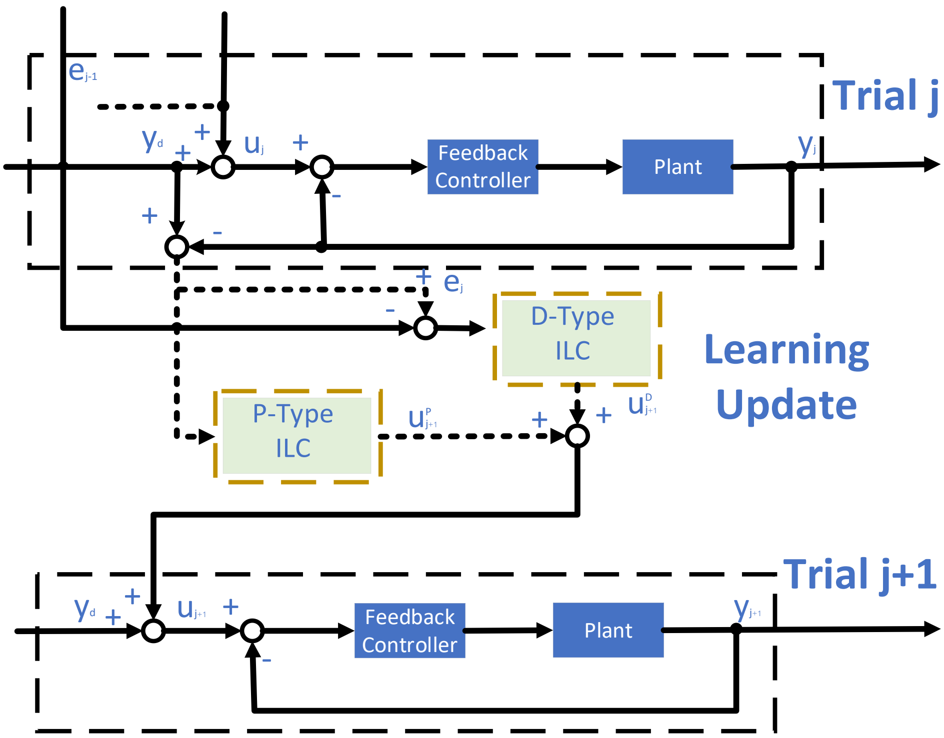


Figure 5 PD-Type ILC.

4. Results and Discussion

The system simulation test was conducted using equations (1), (2), and (3). The system consisted of three tests in the simulation of the joints A, B, and C of the motor.

The simulation had three events to design the controllers, 1) P-Type ILC, 2) D-Type ILC, and 3) PD-Type ILC. In three events, to configure the learning control matrix of the system as shown in Table 1, the number of learning control matrices obtained from the condition of the asymptotic convergence in equation (7) is the eigenvalue condition, and (8) is the singular value condition to show as the Table 2. did not exceed 13.7425, 15.1551, and 16.6739 in motors A, B, and C, respectively, considering the condition of the L matrix in the condition of asymptotic convergence, in which the gain was not more than 5%. The results of the system testing can be discussed as follows.

Table 1 Learning gain values used in the ILC system

Format of ILC	L_P	L_D	note
P-Type ILC	0.50	0.00	
D-Type ILC	0.00	0.10	
PD-Type ILC	0.50	0.10	

Table 2 The convergence condition of the control signal is asymptotically stable

Format of ILC	*Motor A*	*Motor B*	*Motor C*
eigenvalue	0.9636	0.9670	0.9700
singular value	0.9966	0.9969	0.9978

The results of the system test by considering the RMSE are shown in Table 3 and Figure 8. It was found that P-Type ILC, D-Type ILC, and PD-Type ILC had a tendency to work similarly on all 3 joints.

By testing the system, it was noticeable that the PD-Type ILC was designed. The controller had the fastest reduction in RMS error, while the P-Type ILC had similar reductions compared to the PD-Type ILC.

In iterations at 30, the P-type ILC system was able to reduce the RMSE by 98.53% while PD-type ILC could reduce the RMSE by 98.98% from the original, by an average of between 0.44-0.64%, and the RMSE value was approaching zero. The RMSE of the PD-type ILC was a continuous decrease of approximately 7.61%.

In iterations at 400, the P-type ILC system was able to reduce the RMSE by 99.99% while PD-type ILC could reduce the RMSE by 100.00% from the original, and the RMSE value was approaching zero. In the RMSE of D-Type ILC, there was a continuous decrease of approximately 39.57%.

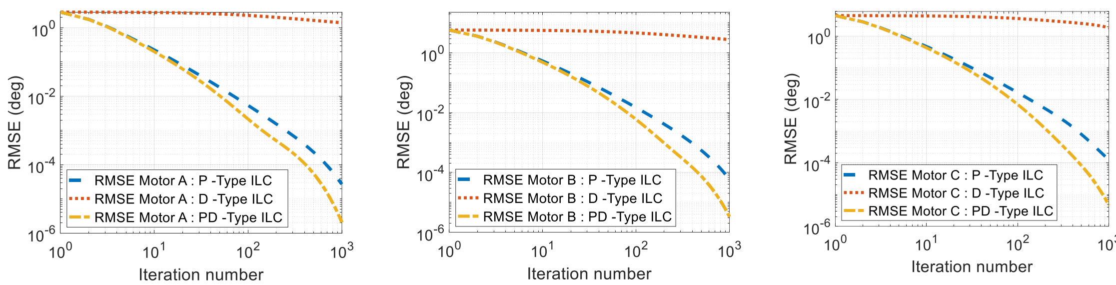
In iterations at 1,000, the P-type ILC system was able to reduce the RMSE by 100.00% while PD-type ILC can reduce the RMSE by 100.00% from the original, and the RMSE value is approaching zero. In the RMSE of D-Type ILC, there was a continuous decrease of approximately 50.09%.

To compare the reduction of the RMSE, a sample RMSE of 8 data from 1,000 data was presented. Table 3 shows the result of the reduction of this RMSE. It was found that this PD-Type ILC control scheme achieved the lowest value throughout the 1,000 iterations and the fastest. This decrease in RMSE was related to Table 2 for the asymptotic convergence. It can be summarized that if the eigenvalue in equation (7) is less than 1, the system can be controlled in P-Type ILC. However, the singular value in equation (8) was less than 1, this system could be confirmed to use P-Type ILC since it could control the stability of the system throughout the next iteration.

This trend showed that gaining more control within the system could help to reduce the RMSE. The trend of ILC control system design results, hence, can be applied in real systems.

Table 3 RMS error values of prototype robotic arm systems in each joint motor

Iteration	A-Joint motor			B-Joint motor			C-Joint motor		
	P-Type	D-Type	PD-Type	P-Type	D-Type	PD-Type	P-Type	D-Type	PD-Type
1	2.8588	2.8588	2.8588	5.7335	5.7335	5.7335	4.4462	4.4462	4.4462
3	1.1460	2.8419	1.1260	2.3990	5.7007	2.3601	1.9297	4.4217	1.9004
5	0.5941	2.8252	0.5678	1.3039	5.6686	1.2515	1.0881	4.3975	1.0478
10	0.2253	2.7850	0.2003	0.5321	5.5906	0.4792	0.4711	4.3389	0.4283
100	0.0054	2.2922	0.0021	0.0147	4.6129	0.0059	0.0165	3.5894	0.0070
400	0.0004	1.7275	0.0001	0.0009	3.4588	0.0001	0.0014	2.6753	0.0002
700	9.20E-05	1.5174	1.45E-05	0.0002	3.0215	2.09E-05	0.0003	2.2502	2.43E-05
1000	2.67E-05	1.4035	2.06E-06	5.81E-05	2.7524	3.32E-06	0.0001	1.8544	4.59E-06



joint of motor A | joint of motor B | joint of motor C

Figure 6 RMSE in joint of motor A, B, and C

From Figures 7, 8, and 9, the system test results showed the tracking performance in each of the ILC iterations in which the researcher has shown the actual path through three samples. The actual path consisted of the first iteration, third iteration, and one hundred iterations. It was performed by the P-Type ILC in Figure 7 and the PD-Type ILC in Figure 9, which was the same, while the PD-Type ILC was slightly better than P-Type ILC. According to the D-Type ILC controller in Figure 8, the actual path was so similar that the first iteration and third iteration changed slightly. The output of the system was not satisfactory in one hundred iterations since it did not perform well in a steady state. The RMSEs could not be reduced to the lowest value. According to the output of the system overview of P-Type ILC and D-Type ILC, it was found that the P-Type ILC was able to adjust the error values quickly, which was similar to P control in the PID system, while D-Type ILC slowed the adjustment, which allowed a positive adjustment of small error values and a reduction in the occurrence of overshoot. It could be reduced in the ILC and was similar to D control in the PID system. Therefore, the PD-Type ILC combined the two systems to optimize the efficiency of reducing the RMSE. It was found that the system had a slight improvement in performance by using the D-Type ILC.

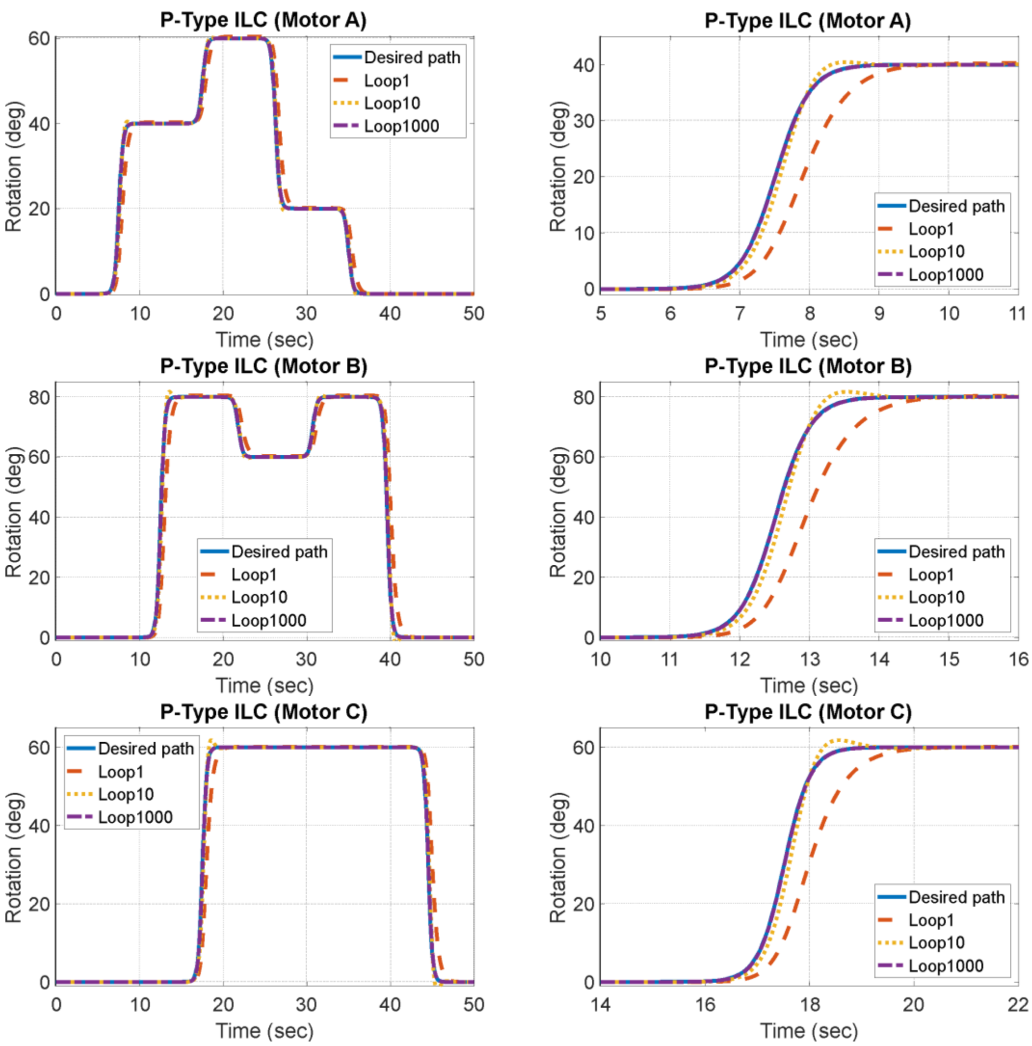


Figure 7 System response results in P-Type ILC.

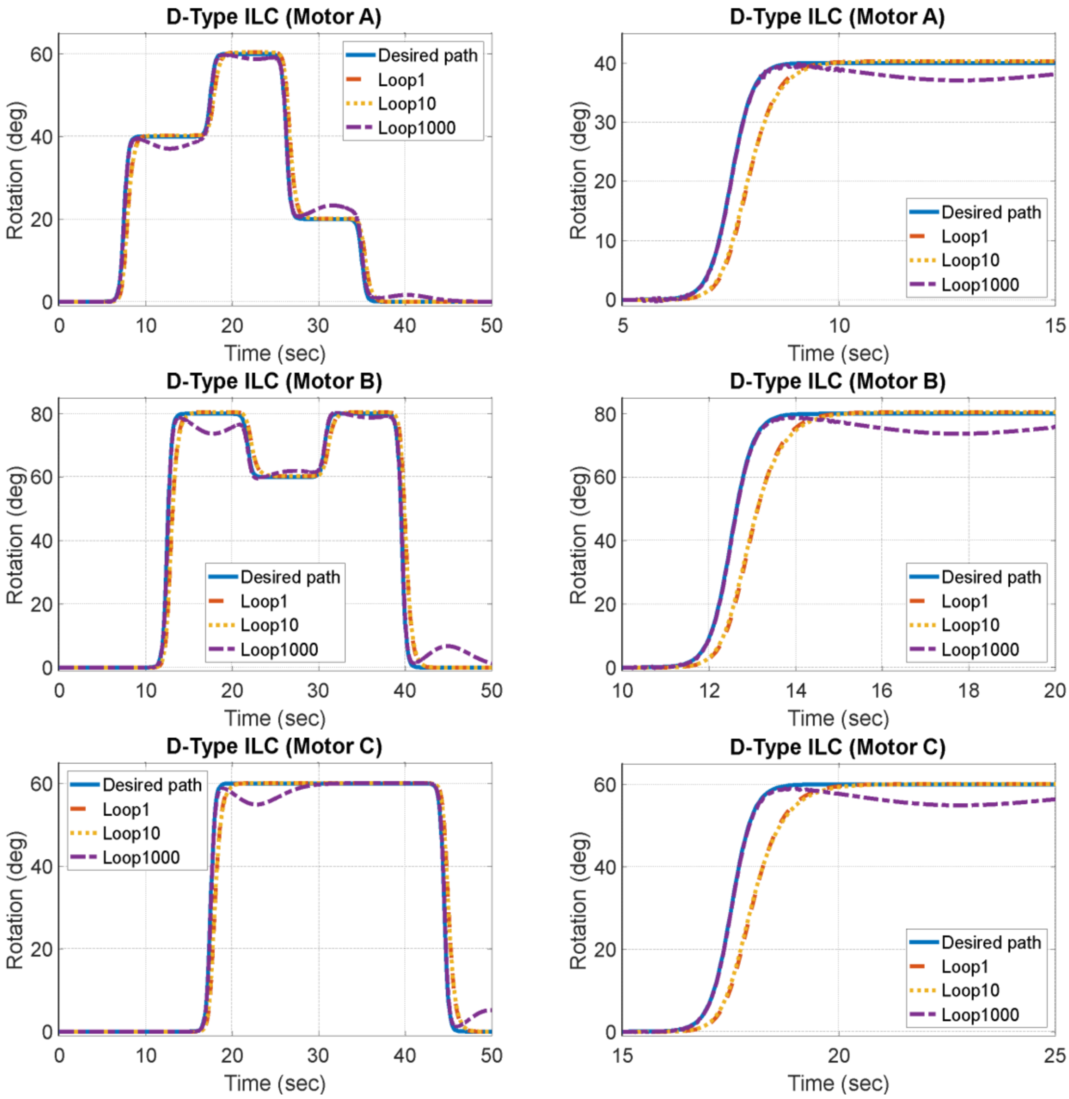


Figure 8 System response results in D-Type ILC.

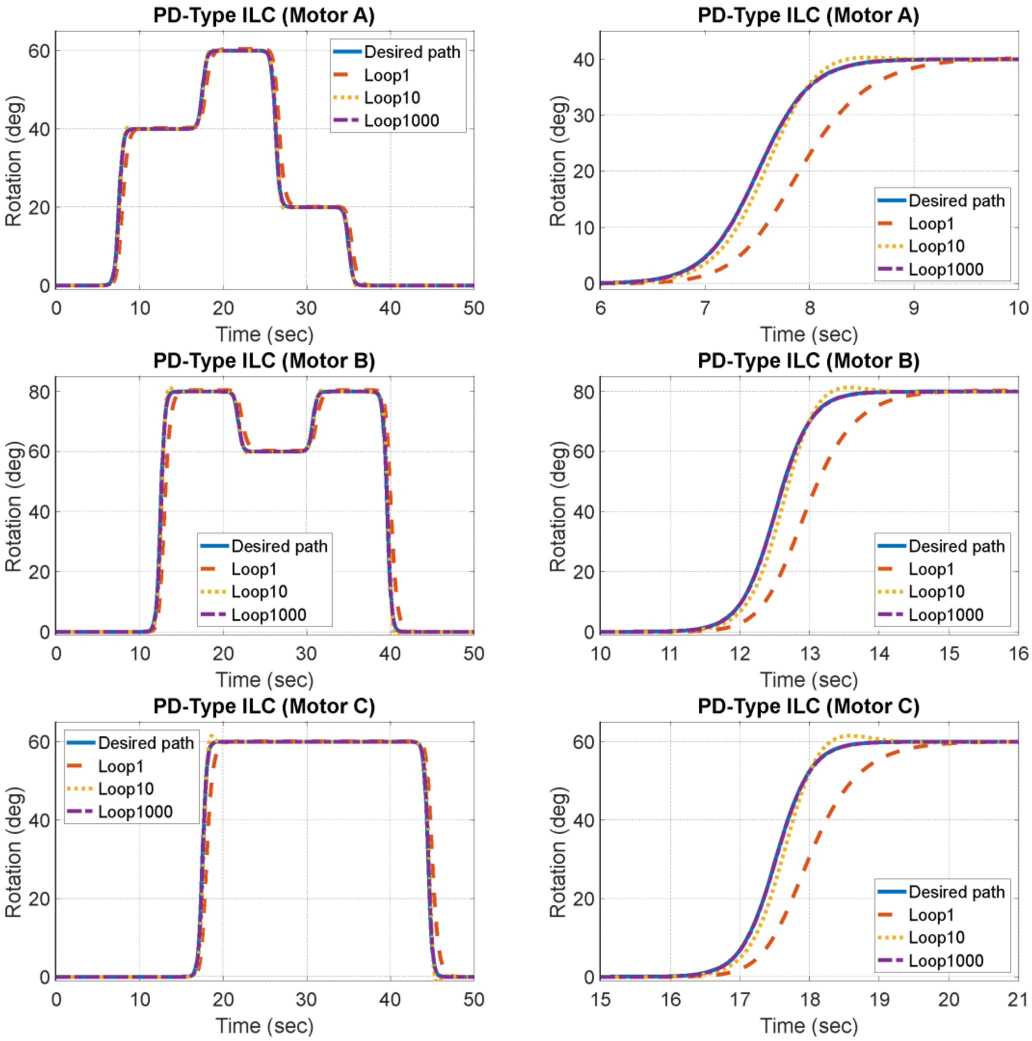


Figure 9 System response results in PD-Type ILC

5. Conclusion

In this research, the researcher was interested in studying the fundamental behavior of different ILC types. The simulation control of a robotic arm showed the basic operating characteristics of the ILC control system in different types. Two factors, 1) The value of the learning control matrix and 2) Overshoot or magnitude of error in the system, were considered to affect the decrease in RMSE occurring within the system. This research considered the second factor of different error values in P-type ILC and D-type ILC. The difference in the error value in each iteration resulted in the reduction the RMSE within the system. PD-type ILC, which had better performance than P-type ILC, was due to the learning control matrix of the system and error variables resulting in a more suitable control input estimation in this system. In the future, the development of this ILC control system may use a time-varying design or a high-order ILC technique to improve the reduction of RMSE values instead of using a fixed gain ILC.

6. Acknowledgements

The researcher would like to thank the Research Institute, Rangsit University for the grant of a research funding to the research team to create a prototype robotic arm.

7. References

Ahn, H. S., Chen, Y., & Moore, K. L. (2007). Iterative learning control: Brief survey and categorization. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 37*(6), 1099-1121.

Bristow, D. A., Tharayil, M., & Alleyne, A. G. (2006). A survey of iterative learning control. *IEEE control systems magazine, 26*(3), 96-114.

Chen, Y., & Moore, K. L. (2002, October). An optimal design of PD-type iterative learning control with monotonic convergence. *In Proceedings of the IEEE Internatinal Symposium on Intelligent Control* (pp. 55-60). Vancouver, BC, Canada.

Cupertino, F., Giordano, V., Naso, D., & Delfine, L. (2006). Fuzzy control of a mobile robot. *IEEE robotics & automation magazine, 13*(4), 74-81.

Dexu, B., Weiwei, K., & Yunlong, Q. (2019). A task-space tracking control approach for duct cleaning robot based on fuzzy wavelet neural network. *Journal of Dynamic Systems, Measurement, and Control*, *141*(11), 1-11.

Ferrara, A., Incremona, G. P., & Sangiovanni, B. (2019). Tracking control via switched Integral Sliding Mode with application to robot manipulators. *Control Engineering Practice*, *90*, 257-266.

Liu, H., Brown, D. J., & Coghill, G. M. (2008). Fuzzy qualitative robot kinematics. *IEEE Transactions on Fuzzy Systems*, *16*(3), 808-822.

Madady, A. (2008). PID type iterative learning control with optimal gains. *International Journal of Control, Automation, and Systems*, *6*(2), 194-203.

Mason, S., Righetti, L., & Schaal, S. (2014, November). Full dynamics LQR control of a humanoid robot: An experimental study on balancing and squatting. *In 2014 IEEE-RAS International Conference on Humanoid Robots* (pp. 374-379). Madrid, Spain.

Moore, K. L., Chen, Y., & Ahn, H. S. (2006, December). Iterative learning control: A tutorial and big picture view. *In Proceedings of the 45th IEEE Conference on Decision and Control* (pp. 2352-2357). San Diego, CA, USA.

Pan, Y., Li, X., & Yu, H. (2018). Efficient PID tracking control of robotic manipulators driven by compliant actuators. *IEEE Transactions on Control Systems Technology*, *27*(2), 915-922.

Song, Z., Mao, J., & Dai, S. (2005). First-order D-type iterative learning control for nonlinear systems with unknown relative degree. *Acta Automatica Sinica*, *31*(4), 555.

Sun, L., & Gan, J. (2010, May). Researching of two-wheeled self-balancing robot base on LQR combined with PID. *In 2010 2nd International Workshop on Intelligent Systems and Applications* (pp. 1-5). Wuhan, China.

Van Den Berg, J., Abbeel, P., & Goldberg, K. (2011). LQG-MP: Optimized path planning for robots with motion uncertainty and imperfect state information. *The International Journal of Robotics Research*, *30*(7), 895-913.

Wang, D. (2000). On D-type and P-type ILC designs and anticipatory approach. *International Journal of Control*, *73*(10), 890-901.