

PID-Type Iterative Learning Control for Output Tracking Gearing Transmission Systems

Luong Thuy Anh ^a, Tran Thi Thanh Nga ^{b,*}, Vu Van Hoc ^b

^a Vietnam - Korea Vocational College of Technology, Bac Giang 220000, Vietnam

^b Thai Nguyen University of Technology, Thai Nguyen 250000, Vietnam

¹ nathanhtnut@gmail.com

* Corresponding Author

ARTICLE INFO

Article history

Received 02 July 2021

Revised 16 July 2021

Accepted 31 July 2021

Keywords

Machine Learning Control;

Controller;

Iterative Learning Algorithm;

PID

ABSTRACT

In this paper, we propose a modified version of the Proportional Integral Derivative (PID)-type iterative learning algorithm. It is very simple to implement on a digital control device for tracking control a continuous-time system. Matlab software is used to model and simulate control algorithms. The simulative application of it to control a gearing transmission system, such that its output response follows the desired trajectory, is then carried out computationally. Obtained studying results proves that this proposed iterative learning algorithm has provided a good output tracking behavior as expected and which is robust in the sense of reducing external disturbance effects.

This is an open-access article under the [CC-BY-SA](#) license.



1. Introduction

Gearing transmission systems (GTS) are the most important mechanical devices in the manufactured industry. They are met everywhere in the industrial world. However, even till now, the problem of output tracking them to the desired trajectory in the presence of inevitable vibrations and noises is still not fulfilled perfectly. The reason is that the vibrations and noises affect the dynamic of the GTS heavily, and in order to improve the control behavior in the sense of reducing these effects by using an existing robust and adaptive conventional control method, it is needed compulsorily a sufficiently precise mathematical model of GTS, which is suitable for this applied method. There have been many studies on the mathematical model of GTS [1]-[6], but it is difficult to get an exact model.

In this situation, the Iterative learning control (ILC) is considered the optimal option because ILC does not use an exact mathematical model for controller design [7][8]. ILC is an intelligent control algorithm. The controller has the ability to learn, memorize and improve quality. ILC was first proposed by Uchiyama in 1978 [9] and lightly developed afterward by Arimoto, Kawamura, Miyazaki in Japan during the 1980s [9][10]. Until this new control concept is widely known outside Japan, it has been really upgraded. Since then, the published works focus mainly on the stability analysis of ILC systems [10]-[14] and in the extension of it into nonlinear ILC [12]-[14], optimal ILC, repetitive learning [13]-[15], adaptive ILC [15]-[17]. Some fields have successfully applied ILC, such as industrial robots [18][19], computer numerical control (CNC) machine tools [20], wafer stage motion systems [21], injection-

molding machines [22], chain conveyor systems [23], rapid thermal processing [24]. ILC has also found application to systems that do not have identical repetition. For instance, in [25], an underwater robot uses similar motions in all of its tasks but with different task-dependent speeds. These motions are equalized by a time-scale transformation, and a single ILC is employed for all motions. ILC can also serve as a training mechanism for open-loop control. This technique is used in [26] for fast-indexed motion control of low-cost, highly nonlinear actuators. As part of an identification procedure, ILC is used in [27] to obtain the aerodynamic drag coefficient for a projectile. Finally, [28] proposes the use of ILC to develop high-peak-power microwave tubes. Meaningful overviews and applications of the ILC development could be found in [29]-[43].

ILC works on the principle of remembering the input-output signals, comparing them with the reference signal, and adjusting the input signal so that the output signal is close to the reference signal. Results in the next iteration are better than the previous one. This process is repeated until the required quality is achieved. Therefore, ILC is called “intelligent” and “learning” [12].

This work presented an ILC algorithm, which is applied to output tracking the GTS with a given desired tracking performance. The results are demonstrated by several different simulation scenarios.

The organization of this paper is as follows. First, in Section 2, the stability of GTS will be studied to clarify whether the ILC concept could be applied directly to GTS or a conventional controller has to be utilized additionally. Then, in Session 3, the proposed discrete-time version of the PID-type ILC algorithm for output tracking the GTS computationally is presented. Here also the detailed control procedure for implementing this ILC version is exhibited. Next, in Section 4, the simulations on GTS will be carried out, and based on obtained results, the tracking stability of the closed-loop sampled-data system, as well as its robustness, will be discussed. Finally, some conclusions and remarks are given in Section 5.

2. Stability Analysis of GTS

It is well known that in the ILC concept, the mathematical models of a controlled process are not often required for controller design [9][12][13]. But the necessary condition for a successful application of ILC is that the controlled process must be stable. In the circumstances that the controlled process is initially unstable, it must be stabilized first by using a suitably conventional technique [9][11][13][31]. Hence, for the success of a direct application of ILC to output tracking control the GTS, the stability analysis of GTS has to be carried out primarily, which is obligatory to require its mathematical model.

The mathematical model of GTS with backlash was already presented in detail in [3][5][45] as follows:

$$\begin{cases} \bar{J}_1 \ddot{\varphi}_1 + \hat{c} r_{L1}^2 \cos^2 \alpha_L (\varphi_1 + i_{12} \varphi_2) = M_d - M_{ms1} \\ \bar{J}_2 \ddot{\varphi}_2 + \hat{c} r_{L2}^2 \cos^2 \alpha_L (\varphi_2 + i_{21} \varphi_1) = -M_c - M_{ms2} \end{cases} \quad (1)$$

where

- $\hat{c} = \begin{cases} c, & \text{for fitting stage} \\ 0, & \text{for backlash phase} \end{cases}$
- c is the stiffness factor of gear material,
- $\bar{J}_1 = J_d + J_1$ with J_d, J_1 are respectively the inertial moment of guided motor and slave drive,
- r_{L1}, r_{L2} are outer radii of the guided gear (Gear1) and the slave gear (Gear2), respectively,
- α_L is the fitting angle between both gears,

- M_d is the desired reference moment, and M_c is the moment of disturbances (load moment),
- M_{ms1} , M_{ms2} are separately the friction moments in Gear1 and in Gear2,
- $i_{12} = i_{12}^{-1} = r_{02} / r_{01}$ is the quotient of inner radii of the gears,
- φ_1 , φ_2 are rotation angles of Gear1 and Gear2, respectively.

Under the assumptions that the friction moments M_{ms1} , M_{ms2} appeared in (1) is depend only on the rotation velocities of the gears:

$$M_{ms1} = \gamma \dot{\varphi}_1 \quad \text{and} \quad M_{ms2} = \gamma \dot{\varphi}_2 \quad (2)$$

and additionally, if both gears (Gear1 and Gear2) are almost fitted to each other so that the system backlash between them could be approximated by:

$$\hat{c} \approx c, \quad \forall t \quad (3)$$

with an input time delay:

$$f(t - \tau) = M_d, \quad (4)$$

where τ denotes the time delay constant, then the general mathematical model (1) given above can be rewritten in:

$$\begin{cases} \bar{J}_1 \ddot{\varphi}_1 + \gamma_1 \dot{\varphi}_1 + c_{z1}(\varphi_1 + i_{12}\varphi_2) = u(t - \tau) \\ \bar{J}_2 \ddot{\varphi}_2 + \gamma_2 \dot{\varphi}_2 - c_{z2}(\varphi_2 + i_{21}\varphi_1) + M_c = 0 \end{cases} \quad (5)$$

where:

$$c_{z1} = cr_{L1}^2 \cos^2 \alpha_L, \quad c_{z2} = cr_{L2}^2 \cos^2 \alpha_L.$$

Furthermore, if the system dynamics is not too fast, that there:

$$\ddot{\varphi}_2 = 0 \quad (6)$$

could be considered, then together with the assumption:

$$M_c = 0 \quad (7)$$

the second equation (5) would be converted in:

$$\begin{cases} c_{z2} i_{21} \varphi_1 = -c_{z2} \varphi_2 + \gamma_2 \dot{\varphi}_2 \\ c_{z2} i_{21} \dot{\varphi}_1 = -c_{z2} \dot{\varphi}_2 \end{cases} \Leftrightarrow c_{z2} i_{21} \begin{pmatrix} \varphi \\ \dot{\varphi}_1 \end{pmatrix} = \begin{pmatrix} -c_{z2} & \gamma_2 \\ 0 & -c_{z2} \end{pmatrix} \begin{pmatrix} \varphi_2 \\ \dot{\varphi}_2 \end{pmatrix}$$

which is obviously coincidental with

$$\begin{pmatrix} \varphi_2 \\ \dot{\varphi}_2 \end{pmatrix} = c_{z2} i_{21} \begin{pmatrix} -c_{z2} & \gamma_2 \\ 0 & -c_{z2} \end{pmatrix}^{-1} \begin{pmatrix} \varphi \\ \dot{\varphi}_1 \end{pmatrix} = \frac{c_{z2} i_{21}}{c_{z2}^2} \begin{pmatrix} -c_{z2} & -\gamma_2 \\ 0 & -c_{z2} \end{pmatrix} \begin{pmatrix} \varphi_1 \\ \dot{\varphi}_1 \end{pmatrix}$$

and thus:

$$\varphi_2 = \beta_1 \varphi_1 + \beta_2 \dot{\varphi}_1 \quad (8)$$

where:

$$\beta_1 = -i_{21} \quad \text{and} \quad \beta_2 = -\frac{c_{z2} i_{21} \gamma_2}{c_{z2}^2}.$$

Now, the substitution of (8) back into the first equation (5) will deduce:

$$\begin{aligned} f(t - \tau) &= \bar{J}_1 \ddot{\varphi}_1 + \gamma_1 \dot{\varphi}_1 + c_{z1}[\varphi_1 + i_{12}(\beta_1 \varphi_1 + \beta_2 \dot{\varphi}_1)] \\ &= \bar{J}_1 \ddot{\varphi}_1 + (\gamma_1 + c_{z1}) \end{aligned} \quad (9)$$

which, simultaneously with the designation $y = \varphi_2$ for system output in (9), implies the following transfer function of GTS finally:

$$G(s) = \frac{Y(s)}{\Phi_1(s)} \cdot \frac{\Phi_1(s)}{F(s)} = \frac{\beta_1 + \beta_2 s}{\alpha_1 s^2 + \alpha_2 s + \alpha_3} e^{-\tau s} \quad (10)$$

where $Y(s)$, $\Phi_1(s)$, $F(s)$ denote Laplace transfers of $y(t)$, $\varphi_1(t)$, $f(t)$, respectively, and all system parameters in it

$$\alpha_1 = \bar{J}_1, \alpha_2 = \gamma_1 + c_{z1} i_{12} \beta_2, \alpha_3 = c_{z1} (1 + i_{12} \beta_1) \quad (11)$$

are positive [44]. Since the positiveness of $\alpha_1, \alpha_2, \alpha_3$ the transfer function $G(s)$ given in (10) is stable and hence the GTS described in (1), with a few of practically acceptable assumptions (2)-(4), (6) and (7), is also stable. Therefore, an ILC algorithm could be applied directly to GTS without using additionally any conventional controller.

3. Iterative learning controller design

As discussed in Section 1, iterative learning controllers work on the principle of “memorizing,” “learning,” and “improving.” The working principle is shown in Fig. 1 and Fig. 2.

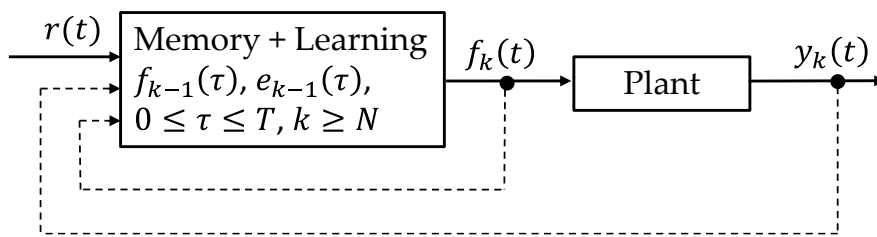


Fig. 1. Block diagram of ILC

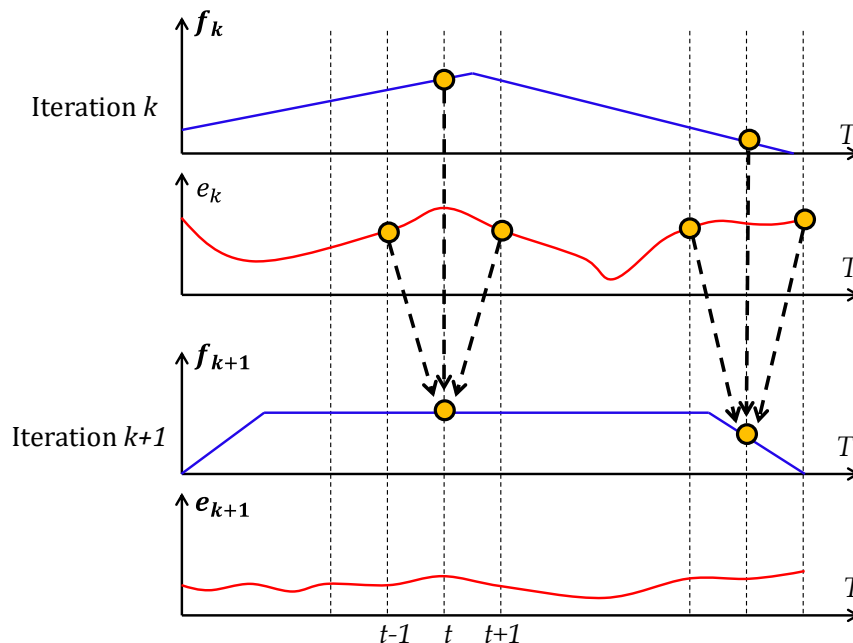


Fig. 2. Working principles of ILC

The most important goal of the algorithm is to define a learning function (also called the learning algorithm [11][12][15]):

$$f_{k+1}(t) = L(f_k(\tau_1), e_k(\tau_2), t) \text{ where } 0 \leq \tau_1, \tau_2 \leq T \quad (12)$$

for controlled systems, described by $P: f(t) \mapsto y(t)$, The work in a repeated manner during a finite time interval $0 \leq t \leq T$, so that the output tracking will be satisfied (13) after a finite trial number $k \geq K$, in which the trial number by learning is subscript k indicates.

$$\|e_k(t)\| < \varepsilon \quad (13)$$

with any constant $\varepsilon \geq 0$, and

$$e_k(t) = r(t) - y_k(t) \quad (14)$$

is the tracking error belongs to the k^{th} trial, $r(t)$ is the desired reference for controlled system $P: f(t) \mapsto y(t)$. GTS is a stable system and operates periodically. Therefore output tracking can be controlled directly by the algorithm (12) [29].

The principle of the ILC algorithm (12) can be explained as follows:

- First, in the process of the k^{th} trial, the input signals $u_k(t)$, $0 \leq t \leq T$ are used to control the plant for producing its complete response $y_k(t)$, $0 \leq t \leq T$.
- After that, at the moment when the current trial is finished, the controller calculates the output tracking error (14), bases on a learning algorithm (12), and determines the input signal $f_{k+1}(t)$, $0 \leq t \leq T$ of the next $(k+1)^{\text{th}}$ trial, In after trial has to produce a smaller tracking error than the previous input $f_k(t)$. more specifically, it must frequently satisfy:

$$\|e_{k+1}(t)\| < \|e_k(t)\| \quad (15)$$

For $k = 0, 1, K$. Additionally, it is noting here that in all trials, the controller parameters the initial values $P: f(t) \mapsto y(t)$ should be the same.

- Finally, the learning process will be continued repetitively $k = 0, 1, \dots$, until the condition (13) is satisfied.

We can also apply a useful version algorithm (12), without having to check the required terminative condition (13), is the following uninterrupted one:

$$f_{k+1}(t) = f_k(t) + l(e_k(\tau)) \quad (16)$$

where $0 \leq \tau \leq T$ and $l(e_k(\tau)) = 0$ if $\|e_k(\tau)\| < \varepsilon$ or if $e_k(\tau) = 0$. As opposed to the ILC algorithm (12), this uninterrupted learning algorithm allows the learning process to faster and being continued forever without any degradation.

In the next part, we design a controller for GTS according to the ILC algorithm (12) or (16). Since the first time the ILC learning algorithm (12) is proposed by Uchiyama [9], ILC has been researched and developed by many authors with many important results, which having been published in several works [9]-[17]. A few important summaries of ILC development are introduced at [29]-[31], in which also a few of the following practicably powerful linear ILC algorithms of the alternative structure (16) are given (see [9][10] for more examples):

$$f_{k+1}(t) = f_k(t) + K e_k(t) \quad (17)$$

or [12][13][16][17]:

$$f_{k+1}(t) = f(t) + K e_k^{(i)}(t), \quad (18)$$

and

$$f_{k+1}(t) = f_k(t) + K_1 e_k(t) + K_2 \dot{e}_k(t), \quad 0 \leq t \leq T, \quad (19)$$

The equation (20) is called Derivative (D-type) algorithm and the equation (19) is Proportional Derivative (PD-type) algorithm. One of the most general linear ILC algorithms (16) is in [10] introduced PID-type algorithm as follows:

$$f_{k+1}(t) = f_k(t) + K_1 e_k(t) + K_2 \int_0^t e_k(\tau) d\tau + K_3 \dot{e}_k(t) \quad (21)$$

Of course, all learning parameters K , K_1 , K_2 and K_3 of these ILC algorithms above have to be determined accordingly to the convergence condition (13) or (15).

In this paper, the PID-type algorithm (21) will be applied to design the iterative learning controller for output tracking control of the GTS. In order that this PID-type algorithm should become more conveniently for computational implementation, the original PID-type algorithm (21) above will be here modified into a so-called sampled data ILC algorithm as follows:

$$f_{k+1}(i) = \begin{cases} f_k(i) + (K_1 + K_2)e_k(i) + K_3 e_k(i+1) & \text{for } i = 0 \\ f_k(i) + K_1 e_k(i-1) + K_2 e_k(i) + K_3 e_k(i+1) & \text{for } 1 \leq i \leq N-1 \\ f_k(i) + K_1 e_k(i-1) + (K_2 + K_3)e_k(i) & \text{for } i = N. \end{cases} \quad (22)$$

After that, for being applicable to control a continuous-time system, the obtained discrete-time values $f_k(i)$, $i = 1, 2, \dots, N$ of this sampled data ILC algorithm will be sent to the controlled GTS through a ZOH device. It means that the real control input $f_k(t)$ in k^{th} trial for the GTS will be constructed from $f_k(i)$, $i = 1, 2, \dots, N$ in following sense:

$$f(t) = f_k(i) \text{ for } iT_a \leq t < (i+1)T_a, \quad (23)$$

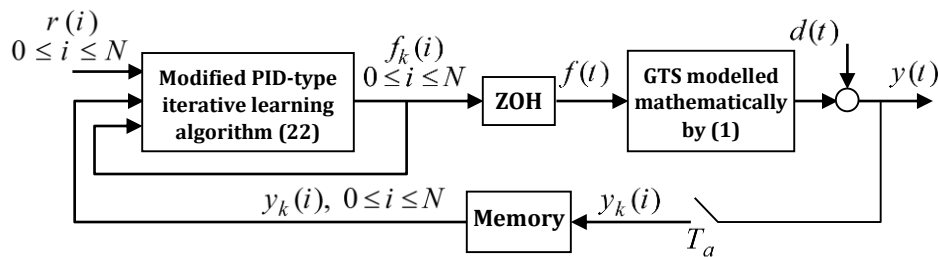


Fig. 3. Sampled data control system with controlled continuous-time GTS and modified discrete-time PID-type ILC.

where $NT_a = T$ and T_a is a chosen sampling time. In this control scheme, the closed-loop control system exhibited in Fig. 3, including the continuous-time GTS and the discrete-time controller, which is established based on the modified PID-type learning algorithm (22), will become an iterative learning sampled data system (in systems science, a sampled data system is defined as a control system, in which a continuous-time plant is controlled by a discrete-time regulator).

The following procedure summarizes all calculating steps above of the introduced discrete-time ILC controller based on the modified learning algorithm (22) to output tracking control the continuous-time GTS.

Control procedure based on modified PID-type learning algorithm

1. Initializing the GTS. Set $f(t) = r(t)$. Choose an appropriate sampling time T_a and then calculate $N = T/T_a$.
2. Determine learning parameters K_1, K_2, K_3 according to the convergent condition (13).
3. Start the k^{th} trial by sending $f(t)$, $0 \leq t \leq T$ it to the controlled GTS.
4. Measure the output values $y_k(i) = y(iT_a)$ and store them successively in the memory until the trial is over, i.e., until the whole $y_k(i)$ for $i = 0, 1, \dots, N$ has been completely

stored in the memory.

5. Use the modified PID-type learning algorithm (22) for calculating $f_{k+1}(i)$, $0 \leq i \leq N$, and then, according to Eq. (23), the whole new input $f(t)$, $0 \leq t \leq T$ is constructed for the next trial.
6. Reset the controlled plant (GTS) to the previous initial values and then return back to step 2, if K_1, K_2, K_3 it has to be determined optimality, which is often different for each trial, or to step 3 otherwise.

4. Simulations and discussions

Hereafter all simulations are carried out based on the control approach illustrated in Fig. 1, where the inputs and outputs of nonlinear GTS during the current iteration (trial) are stored in memory first for determining corrections to GTS inputs accordingly to the modified ILC algorithm (22) in the next iteration. The external signal $d(t)$ in Fig. 1 will be considered here as a repetitive disturbance for representing all system information, which could not be captured entirely in the simulation.

All parameters of GTS, which will be used in simulations, are given in Table 1. Furthermore, to emphasize the advantage of the proposed discrete PID-type ILC algorithm given in (22), the original PD-type ILC algorithm presented (19) is also applied simultaneously for comparison purposes.

Table 1. Simulation parameters

Parameters	Sym.	Value
Stiffness factor of gear material	c	10 [N]
Inertial moment of guided motor	J_1	0.02 [$kg \cdot m^2$]
Inertial moment of the slave drive	J_d	0.01 [$kg \cdot m^2$]
The fitting angle between guided and slave gears	α_L	30^0
The quotient of inner radii of the gears	i_{12}	2
Friction factor in guided motor	γ_1	0.07
Friction factor in slave drive	γ_2	0.07
The outer radius of guided gear (Gear1)	r_{L1}	50 [mm]
The outer radius of slave gear (Gear2)	r_{L2}	100 [mm]
Periodic sampling time	T_a	0.05 [sec]
Parameters of modified PID-type ILC algorithm	K_1	0.1
	K_2	0.3
	K_3	0.3
Parameters of original PD-type ILC algorithm	K_1	0.1
	K_2	0.3
Initial values of GTS	y_0, y_1	0
Operating time interval of GTS	T	10 [sec]

Obtained simulation results without external disturbance $d(t)$ are depicted separately for different references in Fig. 4-Fig. 7. The simulation results, where the external disturbance $d(t)$ occurs repetitively during the learning process, are exhibited in Fig. 8 and Fig. 9.

In Fig. 4 and Fig. 5 the tracking results by trapezoidal reference after 5 and 15 learning trials are illustrated, respectively, whereas Fig. 6 and Fig. 7 exhibit the results with sinusoidal reference also after 5 and 15 learning iterations.

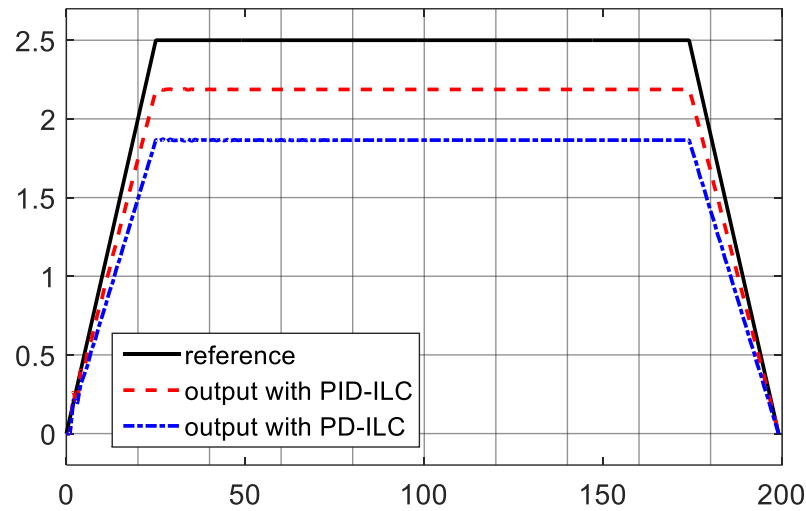


Fig. 4. Trapezoidal tracking error after 5 iterations

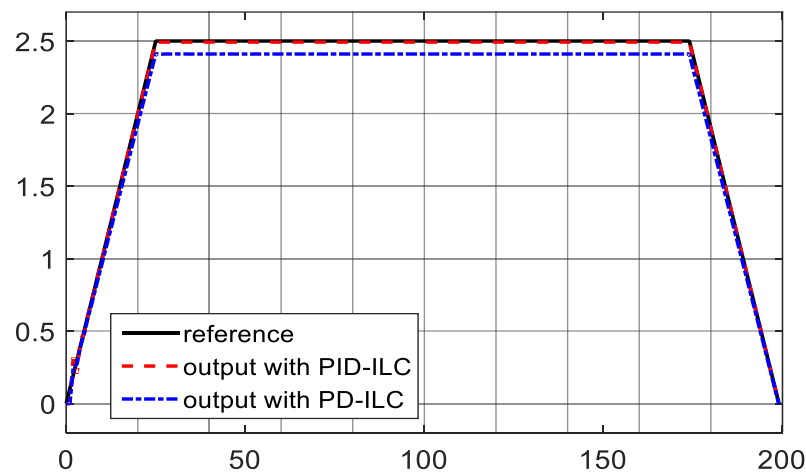


Fig. 5. Trapezoidal tracking error after 15 iterations.

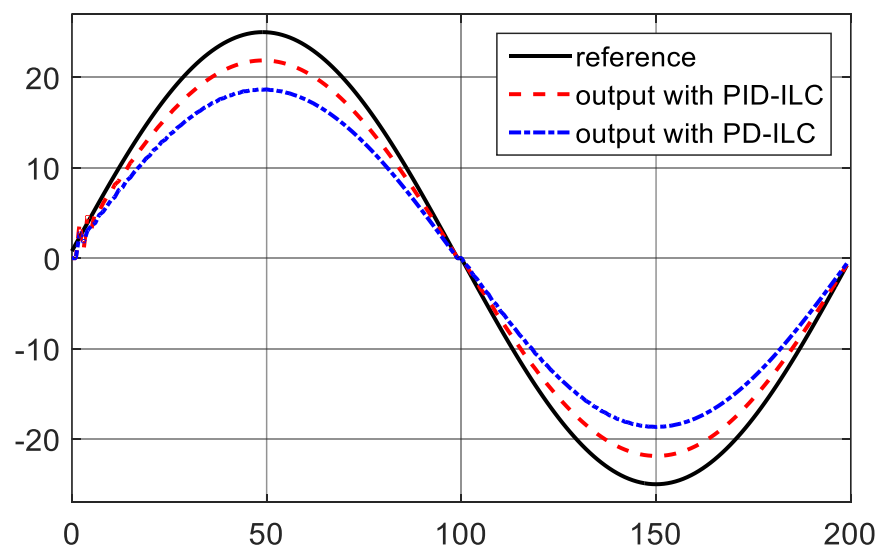


Fig. 6. Sinusoidal tracking error after 5 iterations (trials).

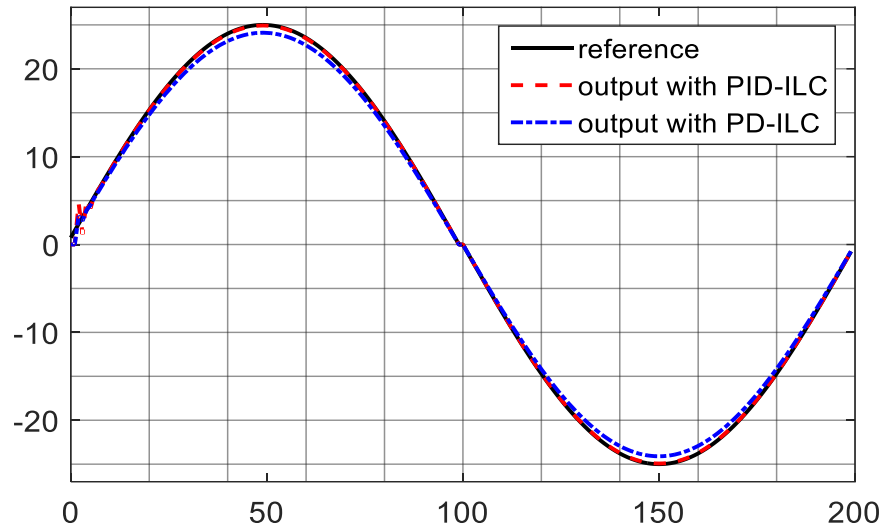


Fig. 7. Sinusoidal tracking error after 15 iterations (trials).

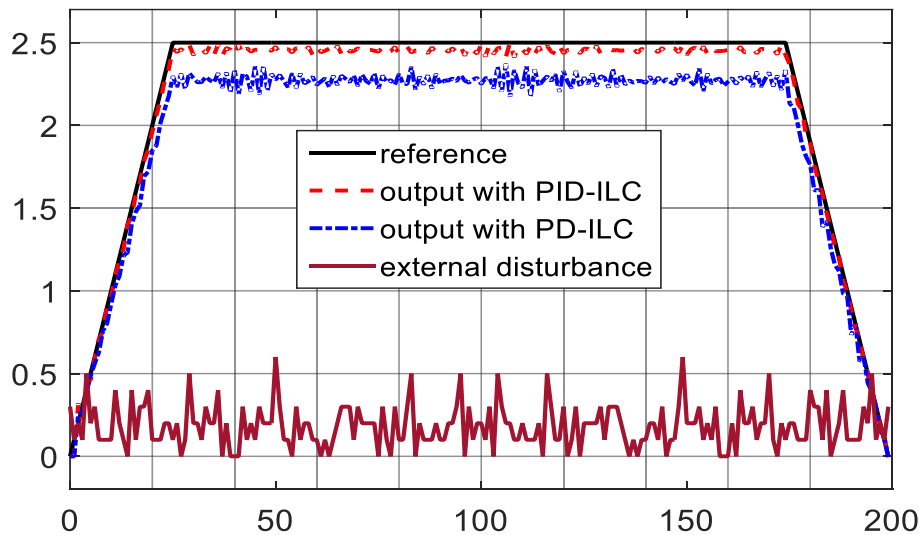


Fig. 8. Disturbance reduction after 10 iterations.

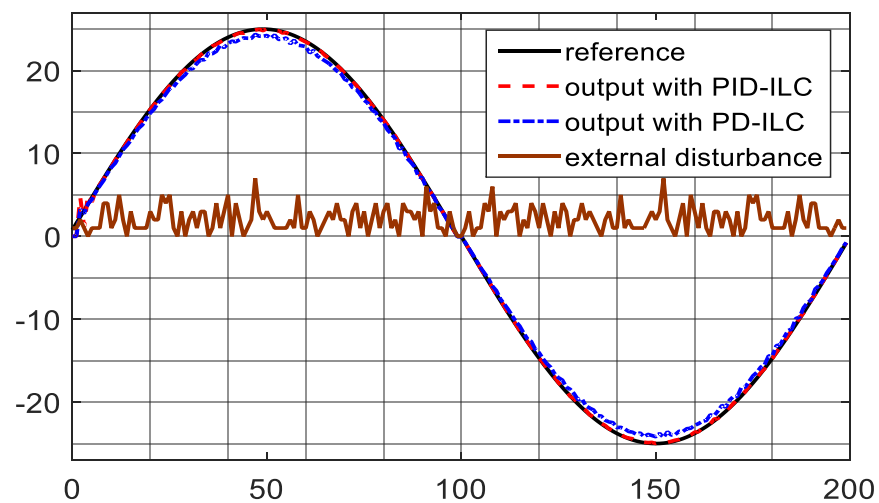


Fig. 9. Disturbance reduction after 20 iterations.

As apprehended from obtained simulation results without external disturbance (Fig. 4-Fig. 7), both PID-type and PD-type ILC produce the same convergent learning performance in common sense that the more learning iterations occur, the smaller output tracking error will be. However, the proposed PID-type ILC algorithm given in (22) provides a better learning behavior than PD-type (19). Precisely, the tracking error by using PID-ILC after 5 learning iterations is only 0.3194, while it is 0.6422 by PD-ILC, or it is just 0.0085 by modified PID-ILC in comparison with 0.1060 by PD-ILC after 15 iterations.

The simulation results for the circumstance that the external disturbance $d(t)$ occurs repetitively during the learning process are exhibited in Fig. 8 and Fig. 9. Once again, it is shown clearly there that both ILC algorithms deliver the same performance of disturbance suppression. Then, the more learning iterations are executed. The better diminution of external disturbance effects will be. But conversely, for being reached a particular suppression quantity, the modified PID-type ILC (22) needs littler learning iterations than the conventional PD-type (19).

5. Conclusion

The paper proposes a modified sample data PID-type iterative learning algorithm for the control of continuous-time systems. An application of this proposed PID-type ILC algorithm for verifying the output tracking performance of continuous-time systems to the desired reference has been simulated by external disturbances perturbed gearing transmission system.

The control algorithm had been simulated without and with an external disturbance. The simulation results show that the controller is suitable for gearing transmission systems operation and has the expected performance. Moreover, it helped the system working more stable, faster in the calculation, and produces smaller tracking errors.

The improvement of system output tracking performance as well as of the disturbance suppression behavior by using this introduced PID-type ILC algorithm had been confirmed experimentally through simulations. However, a strictly theoretical proof of these conclusions is still open because the analysis of sampled-data systems is related to a rich mathematical structure.

Acknowledgment: This work is partly supported by the Vietnam - Korea Vocational College of Technology and the Thai Nguyen University of Technology, Vietnam.

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

References

- [1] J. Wang, R. Li, and X. Peng, "Survey of nonlinear vibration of gear transmission systems," *Appl. Mech. Rev.*, vol. 56, no. 3, pp. 309-329, 2003. <https://doi.org/10.1115/1.1555660>
- [2] N. T. Dzung, D. P. Nam, and N. Q. Dich, "Modelling and Control Design of a V-Shaped Thermal Actuator System via Partial Derivative Equation Approach," *Proceedings of the 5th International Conference on Mechatronics and Robotics Engineering* 2019, pp. 78-82. <https://doi.org/10.1145/3314493.3314516>
- [3] K. (Stevanović) Hedrih and V. Nikolić-Stanojević, "A Model of Gear Transmission: Fractional Order System Dynamics," *Mathematical Problems in Engineering*, vol. 2010, p. 972873, 2010. <https://doi.org/10.1155/2010/972873>
- [4] Z. Xiao, J. Cao, and Y. Yu, "Mathematical Modeling and Dynamic Analysis of Planetary Gears System with Time-Varying Parameters," *Mathematical Problems in Engineering*, vol. 2020, p. 3185624, 2020. <https://doi.org/10.1155/2020/3185624>
- [5] E. Brassitos and N. Jalili, "Dynamic Model Development and Characterization of Gear Bearing Transmission Systems: Theory and Experiments," *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 4, pp. 1651 – 1661, Aug. 2019. <https://doi.org/10.1109/TMECH.2019.2916333>

-
- [6] A. Zhang, J. Wei, L. Shi, D. Qin, and T. C. Lim, "Modeling and dynamic response of parallel shaft gear transmission in non-inertial system," *Nonlinear Dynamics*, vol. 98, pp. 997–1017, 2019. <https://doi.org/10.1007/s11071-019-05241-w>
- [7] A. Zilouchian and M. Jamshidi (Eds.), *Intelligent control systems using soft computing methodologies*, CRC Press, 2001. <https://doi.org/10.1201/9781420058147>
- [8] P. J. Antsaklis, "Intelligent control," in *Wiley Encyclopedia of Electrical and Electronics Engineering*, vol. 10, pp. 493-503, 1999. <https://doi.org/10.1002/047134608X.W1019>
- [9] M. Uchiyama, "Formation of high speed motion pattern of mechanical arm by trial," *Trans. of Society of Instrumentation and Control Engineer*, vol. 19, no. 5, pp. 706-712, 1978. <https://doi.org/10.9746/sicetr1965.14.706>
- [10] S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering operation dynamic systems by learning: A new control theory for servomechanism or mechatronic systems," *Proceedings of 23th Conference on Decision and Control*, Nevada 1984, pp. 1064-1069. <https://doi.org/10.1109/CDC.1984.272176>
- [11] S. Arimoto, "Mathematical theory of learning control with application to robot control," *Proceedings of 4th Workshop on applications of adaptive systems*, Connecticut, 1985, pp. 379-388. https://doi.org/10.1007/978-1-4757-1895-9_27
- [12] K. L. Moore, M. Dahleh, and S. P. Bhattacharyya, "Iterative learning for trajectory control," *Proceedings of Conference on Decision and Control*, Florida, 1989, pp. 860-865. <https://doi.org/10.1109/CDC.1989.70243>
- [13] K. L. Moore, *Iterative learning control for deterministic systems*, London Springer Verlag, 1993. <https://doi.org/10.1007/978-1-4471-1912-8>
- [14] J. X. Xu and Y. Tan, *Linear and nonlinear iterative learning control*, Springer Verlag, 2003. <https://doi.org/10.1007/3-540-44845-4>
- [15] M. Norrloef, *Iterative learning control: Analysis, Design and Experiment*, Diss., No.653, Linköping University, Sweden 2000. <https://www.researchgate.net/publication/2866895>
- [16] V. Vita, A. Vitas, and G. E. Chatzarakis, "Design, implementation and evaluation of an optimal iterative learning control algorithm," *WSEAS Trans. on Circuits and Systems*, vol. 2, no. 10, pp. 39-48, 2011. <http://www.wseas.us/e-library/transactions/circuits/2011/52-094.pdf>
- [17] S. Tian and X. Zhou, "State tracking algorithm for a class of singular ILC systems," *J. Syst. Sci. Math. Sci.*, vol. 32, no. 6, pp. 731-738, 2012. http://sysmath.com/jweb_xtkxysx/EN/abstract/abstract11902.shtml
- [18] Cha'o-Kuang Chen and Kai-Sheng Li, "Iterative learning control for robotic contouring," *2009 IEEE International Conference on Control and Automation*, IEEE, 2009, pp. 1092-1097. <https://doi.org/10.1109/ICCA.2009.5410311>
- [19] S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering operation of robots by learning," *J. Robot. Syst.*, vol. 1, no. 2, pp. 123–140, 1984. <https://doi.org/10.1002/rob.4620010203>
- [20] D. Kim and S. Kim, "An iterative learning control method with application for CNC machine tools," *IEEE Trans. Ind. Appl.*, vol. 32, pp. 66–72, 1996. <https://doi.org/10.1109/28.485814>
- [21] D. D. Roover and O. H. Bosgra, "Synthesis of robust multivariable iterative learning controllers with application to a wafer stage motion system," *Int. J. Control*, vol. 73, pp. 968–979, 2000. <https://doi.org/10.1080/002071700405923>
- [22] H. Havlicsek and A. Alleyne, "Nonlinear control of an electrohydraulic injection molding machine via iterative adaptive learning," *IEEE/ASME Trans. Mechatron.*, vol. 4, pp. 312–323, 1999. <https://doi.org/10.1109/3516.789689>
- [23] A. D. Barton, P. L. Lewin, and D. J. Brown, "Practical implementation of a real-time iterative learning position controller," *Int. J. Control*, vol. 73, pp. 992–999, 2000. <https://doi.org/10.1080/002071700405941>
- [24] D. R. Yang, K. S. Lee, H. J. Ahn, and J. H. Lee, "Experimental application of a quadratic optimal iterative learning control method for control of wafer temperature uniformity in rapid thermal
-

- processing," *IEEE Trans. Semicond. Manuf.* 2003, 16, 36–44. <https://doi.org/10.1109/TSM.2002.807740>
- [25] S. Kawamura and N. Sakagami, "Analysis on dynamics of underwater robot manipulators basing on iterative learning control and time-scale transformation," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2002, pp. 1088–1094. <https://doi.org/10.1109/ROBOT.2002.1014688>
- [26] C. V. Giessen, Q. Zou, and S. Devasia, "Inversion-based precision positioning of inertial reaction devices" in *Proc. Amer. Contr. Conf.*, 2004, pp. 3788–3793. <https://doi.org/10.23919/ACC.2004.1384502>
- [27] Y. Chen, C. Wen, J.-X. Xu, and M. Sun, "High-order iterative learning identification of projectile's aerodynamic drag coefficient curve from radar measured velocity data," *IEEE Tran. Contr. Syst. Technol.*, vol. 6, no. 4, pp. 563–570, 1998. <https://doi.org/10.1109/87.701354>
- [28] C.T. Abdallah, V.S. Soulian, and E. Schamiloglu, "Toward "smart tubes" using iterative learning control," *IEEE Trans. Plasma Sci.*, vol. 26, no. 3, pp. 905–911, 1998. <https://doi.org/10.1109/27.700869>
- [29] S. Tian, Q. Liu, X. Dai, and J. Zhang, "A PD type iterative learning control algorithm for singular discrete systems," *Advances in Difference Equations*, 2016, vol. 321 Springer. <https://doi.org/10.1186/s13662-016-1047-4>
- [30] D. H. Owen, N. Amann, and E. Roger, "Iterative learning control, an overview of recent algorithm," *Applied Mathematics and Computer Science*, vol. 5, no. 3, pp. 425–438, 1995. <https://www.sciencedirect.com/topics/engineering/iterative-learning-control>
- [31] D. H. Owens and J. Hatonen, "Iterative learning control: The state of the art," *IFACT Workshop on Periodic Control Systems*, 2004, pp. 51–62. [https://doi.org/10.1016/S1474-6670\(17\)31443-X](https://doi.org/10.1016/S1474-6670(17)31443-X)
- [32] L. Zhou, H. Tao, W. Paszke, V. Stojanovic, and H. Yang, "PD-Type Iterative Learning Control for Uncertain Spatially Interconnected Systems," *Mathematics*, vol. 8, no. 9, p. 1528, 2020. <https://doi.org/10.3390/math8091528>
- [33] H. F. Tao, W. Paszke, E. Rogers, H. Z. Yang, K. Galkowski, "Iterative learning fault-tolerant control for differential time-delay batch processes in finite frequency domains," *J. Process Control*, vol. 56, pp.112–128, 2017. <https://doi.org/10.1016/j.jprocont.2016.12.007>
- [34] B. Sulikowski, K. Gałkowski, A. Kummert, and E. Rogers, "Two-dimensional (2D) systems approach to feedforward/feedback control of a class of spatially interconnected systems," *Int. J. Control*, vol. 91, no. 12, pp. 2780–2791, 2018. <https://doi.org/10.1080/00207179.2018.1522451>
- [35] H. F. Tao, W. Paszke, H. Z. Yang, and K. Gałkowski, "Finite frequency range robust iterative learning control of linear discrete system with multiple time-delays," *J. Franklin Inst.* vol. 356, no. 5, pp. 2690–2708, 2019. <https://doi.org/10.1016/j.jfranklin.2019.01.040>
- [36] H. L. Xu, Z. P. Lin, X. K. Zhai, H. Y. Feng, and X. F. Chen, "Quadratic stability analysis and robust distributed controllers design for uncertain spatially interconnected systems," *J. Franklin Inst.*, vol. 355, no. 16, pp. 7924–7961, 2018. <https://doi.org/10.1016/j.jfranklin.2018.07.013>
- [37] W. Paszke, E. Rogers, and M. Boski, "Repetitive process based design of PD-type iterative learning control laws," In *Proceedings of the Mediterranean Conference on Control and Automation*, Zadar, Croatia, 19–22 June 2018. [10.1109/MED.2018.8442499](https://doi.org/10.1109/MED.2018.8442499)
- [38] Y. Wu, M. Yang, and J. Zhang, "Open-Closed-Loop Iterative Learning Control with the System Correction Term for the Human Soft Tissue Welding Robot in Medicine," *Mathematical Problems in Engineering*, vol. 2020, p. 2458318, 2020. <https://doi.org/10.1155/2020/2458318>
- [39] D. Luo, J. Wang, D. Shen, and M. Fečkan, "Iterative learning control for fractional-order multi-agent systems," *Journal of the Franklin Institute*, vol. 356, no. 12, pp. 6328–6351, 2019. <https://doi.org/10.1016/j.jfranklin.2019.06.001>
- [40] Y. Zhang, J. Liu, and X. Ruan, "Iterative learning control for uncertain nonlinear networked control systems with random packet dropout," *International Journal of Robust and Nonlinear Control*, vol. 29, no. 11, pp. 6328–6351, 2019. <https://doi.org/10.1002/rnc.4568>

-
- [41] A. C. Antoulas, B. Zhu, Q. Zhang, B. York, B. W. O'Malley, and C. C. Dacso, "A novel mathematical method for disclosing oscillations in gene transcription: A comparative study," In: *PloS one*, vol. 13, no. 9, 2018. <https://doi.org/10.1371/journal.pone.0198503>
- [42] L. Blanken and T. Oomen, "Multivariable iterative learning control design procedures: From Decentralized to Centralized, Illustrated on an Industrial printer," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 4, pp. 1534-1541, 2019. <https://doi.org/10.1109/TCST.2019.2903021>
- [43] D. Shen and X. Li, *Iterative Learning Control for Systems with Iteration-Varying Trial Lengths*, Springer, 2019. <https://www.springer.com/gp/book/9789811361357>
- [44] D. A. Bristow, M. Tharayil, and A. G. Alleyne, "A Survey of Iterative Learning Control: A learning-based method for high-performance tracking control," *IEEE control systems magazine*, vol. 26, no. 3, pp. 96-114, 2006. <https://doi.org/10.1109/MCS.2006.1636313>
- [45] L. Tao, D. Tian, S. Tang, X. Wu, and B. Li, "Dynamical Modelling and Simulation of Spur Gears with Flank Pitch Error," *Chinese Journal of Mechanical Engineering*, In Review, 2021. <https://doi.org/10.21203/rs.3.rs-534359/v1>