12th International Conference on Electrical, Electronics and Computer Engineering (IcETRAN), Čačak, 9-12 June 2025

Comparison of Serial and Parallel Implementations of ILC in a Closed-Loop Feedback System

Milan Matijević
Center for Applied Automatic Control
Faculty of Engineering at University of Kragujevac
Kragujevac, Serbia
matijevic@kg.ac.rs &
https://orcid.org/0000-0001-5307-0027

Mihailo Lazarević
Department for Mechanics
Faculty of Mechanical Engineering, University of Belgrade
Belgrade, Serbia
mlazarevic@mas.bg.ac.rs &
https://orcid.org/0000-0002-3326-6636

Vojislav Filipović
Department for Automatic Control
Faculty of Mechanical and Civil Engineering, University of Kragujevac
Kraljevo, Serbia
v.filipovic@mts.rs &
https://orcid.org/0000-0002-9601-5332

Nikola Živković
Department for Robotics
LOLA Institute
Belgrade, Serbia
nikola.zivkovic@li.rs &
https://orcid.org/0000-0002-2276-2933

Dragan Kostić
Enabling Technology Mechatronics
ASMPT Center of Competency
Beuningen, Netherlands
dragan.kostic@asmpt.com &
https://orcid.org/0000-0001-9286-7389

***Abstract*—Iterative Learning Control (ILC) is a data-driven strategy for precise trajectory tracking in systems that operate repetitively under similar reference trajectories, disturbances, and initial conditions. Despite more than three decades of development, some implementation aspects of the learned ILC signals remain underexplored. This paper compares serial and parallel ILC signal implementations within a closed-loop feedback system. Although both approaches have been equally addressed in the literature, our findings show that the serial implementation provides clear advantages in achieving desired performance.**

***Keywords*—Iterative Learning Control (ILC), Serial ILC implementation, Parallel ILC implementation**

I. Introduction

Iterative Learning Control is a data-driven strategy for precise trajectory tracking in systems that operate repetitively under similar reference trajectories, disturbances, and initial conditions [1].

Although the concept of Iterative Learning Control (ILC) is considered relatively recent, its origins can be traced back to the 1970s, with some of the earliest implementations appearing in patents from that period [2]. Unlike traditional controllers, which do not utilize information from previous executions, ILC incorporates errors from prior iterations into its control strategy. This enables more accurate reference trajectory tracking, even when the system model is not fully known or when unknown but repetitive disturbances occur in each cycle.

The primary goal of an ILC controller is to generate a feedforward or a compensation component of the control signal - derived from stored data of previous iterations - that compensates for repeating disturbances and minimizes tracking errors in subsequent executions [3]. In contrast to feedback controllers, which respond to disturbances only after they arise, ILC takes a predictive approach by relying on learned data from earlier iterations [1]. This approach eliminates delays that typically limit the performance of classical controllers, thereby significantly enhancing the system's dynamic behavior [1]. ILC does not require exogenous signals (such as references or disturbances) to be known or measurable in advance - for instance, an external temperature may be measured as a disturbance before it affects the process [1]. However, ILC assumes that such exogenous signals repeat consistently from one iteration to the next, a condition commonly met in motion control systems [4].

The ILC signal functions as a feedforward control signal generated through a convergence process over multiple iterations. The control input for each iteration is calculated based on data from previous iterations, progressively improving the accuracy of control. As such, ILC belongs to the family of data-driven control algorithms and is well-suited for addressing challenges such as nonlinearity, incomplete model knowledge, and minimizing dynamic tracking errors in reference trajectory following [4, 1].

The first monograph in the field of ILC focused on deterministic systems [5], while three decades later, a comprehensive treatment of ILC applications in linear, nonlinear, distributed, and stochastic systems was presented in [6]. The work in [7] explores frequency-domain analysis and synthesis approaches, as well as digital implementations of ILC algorithms in industrial settings. The book [8] addresses ILC in systems where the duration of the reference trajectory is a random variable. Review papers [9, 10] focus on stochastic ILC algorithms and those with incomplete information, respectively. The survey in [2] outlines the development of analysis and synthesis methodologies for ILC systems over the past decades and emphasizes the increasing relevance of ILC in micro- and nano-manufacturing. Review paper [3] classifies existing ILC applications and highlights its adoption in specialized areas such as robotics, process control, semiconductor manufacturing, and bioengineering. In addition, several doctoral dissertations [11–13] offer a comprehensive systematization of previous findings, covering both theoretical and applied aspects of ILC. In [1], the industrial applications, effects, and challenges of using ILC are discussed, illustrated by a generic example of a wirebonder

ISBN 978-86-6200-031-6 ©2025 AM & ETRAN

https://doi.org/10.69994/12Ic25004

machine used in the manufacturing of semiconductor components and computer chips [14]. High productivity and production quality require machines to execute fast, precise, and accurate movements at high accelerations. Motion control is essential for advancing performance, productivity, and reliability, all of which are expected to improve with each new generation of machines. The quality of the design and implementation of ILC significantly contributes to the overall quality of motion control.

This paper presents the synthesis principle of an ILC algorithm and compares two approaches: 1) serial connection of the ILC signal within a closed-loop system, and 2) parallel connection of the ILC signal within the same. Although both approaches are based on the same convergence principles of the ILC signal, which modifies the control variable of the same closed-loop system, it will be shown that the serial connection approach yields superior results. To the best of the authors' knowledge, this conclusion has not yet been reported in the literature.

II. Design Methodology of the ILC Algorithm

Fig. 1. and Fig. 2. presents the integration of the ILC signal into a feedback control system, where the index j denotes the j-th repetition or iteration of system operation. After each iteration j, the ILC signal f_{j+1} is updated for the subsequent iteration $j+1$, using data from the previous cycle. Through this iterative process, the ILC signal is progressively refined, leading to improvements in the control input and overall system performance.

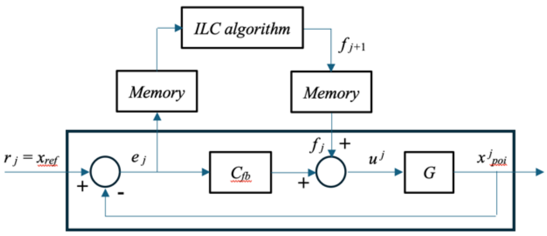

Fig. 1. Parallel integration of the ILC signal f_j into a feedback control system

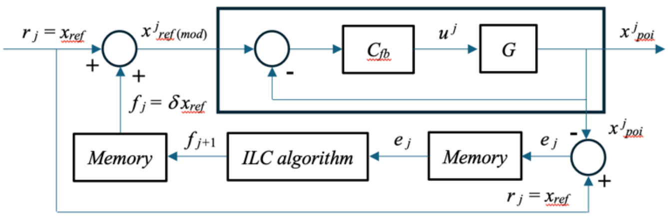

Fig. 2. Serial integration of the ILC signal f_j into a feedback control system

The ILC system operates in two domains: the time (or frequency) domain and the iteration domain. The time or frequency domain captures the dynamics of the plant (G) and the behavior of the feedback control system, as illustrated by the boxed section in Fig. 1 and Fig. 2. The iteration domain describes the convergence of the ILC signal f_{j+1} and the evolution of system behavior across iterations, by monitoring the complete set of measured signal sequences during the j-th iteration such as the tracking error signal e_j. This domain operates in an offline manner with respect to the system's dynamic execution, meaning that the ILC signal f_{j+1} is computed between iterations j and $j+1$ [1].

The fundamental problem in ILC is to design an update algorithm (for calculation of the ILC signal f_{j+1} in iteration $(j+1)$) such that the tracking error signal (e_{j+1}) of the closed-loop system is minimised in some appropriate sense. A general updating equation is given by

$$f_{j+1}(z) = Q(z)\big(f_j(z) + L(z)E_j(z)\big). \quad (1)$$

where the discrete transfer function $L(z)$ is the learning filter, and $Q(z)$ is the robustness filter. Since the computation of relation (1) is performed offline, and the signal sequences f_j and e_j are fully known prior to applying relation (1), non-causal filtering of f_j and e_j using the digital filters $Q(z)$ and $L(z)$ is possible [1]. Typically, $Q(z)$ is a low-pass filter.

For the structure of the feedback control system (the boxed section in Fig. 1 and Fig. 2), the sensitivity function $S(z)$, the process sensitivity function $PS(z)$, and the complementary sensitivity function $T_{cs}(z)$ are defined as follows:

$$S(z) = \frac{1}{1 + C_{fb}(z)G(z)}$$
$$PS(z) = S(z)G(z) \quad (2)$$
$$T_{cs}(z) = \frac{C_{fb}(z)G(z)}{1 + C_{fb}(z)G(z)}$$

where $C_{fb}(z)$ and $G(z)$ are the discrete transfer functions of the feedback controller and the plant, respectively. To ensure a uniform analysis for both the parallel and serial integration of the ILC algorithm in the system shown in Fig. 1, we introduce the substitution:

$$X(z) = PS(z), \text{ for the parallel integration of the ILC} \quad (3)$$

$$X(z) = T_{cs}(z), \text{ for the serial integration of the ILC} \quad (4)$$

We now derive the equation for the propagation of the tracking error from one iteration to the next:

$$E_{j+1}(z) = Q(z)[1 - X(z)L(z)]\, E_j(z) \quad (5)$$

In accordance with equations (5), (4) and (3), the tracking error converges monotonically if and only if [1]:

$$|Q(z)[1 - X(z)L(z)]| < 1 \quad (6)$$

for all frequencies f_h, where $z = e^{i2\pi f_h T_s}$, and T_s is sampling time. From (5), the learning filter $L(z)$ is calculated as the inverse dynamic model of $PS(z)$ or $T_{cs}(z)$ (see (3) and (4)):

$$L(z) \approx (X(z))^{-1} \quad (7)$$

For small sampling times, discrete models often become non-minimum phase (i.e., they have one or more unstable zeros). In such cases, calculating the learning filter $L(z)$ can be challenging [15, 16]. Common approximate methods for computing the filter in (7), including the popular ZPETC method [17], are discussed in detail in [16].

If exact computation of the learning filter (7) is not feasible—whether it is causal or not—the convergence condition (6) must be verified for $Q(z) \equiv 1$. If the condition is not satisfied, a robustness filter ($Q(z) \not\equiv 1$) should be designed to satisfy condition (6). Using graphical-analytical analysis of condition (6) for $Q(z) \equiv 1$, the maximum frequency bandwidth and minimum selectivity required for the robustness filter $Q(z)$ can be determined [1, 16].

III. Comparison of the Effects of Serial and Parallel ILC Algorithms

Contemporary industrial applications of ILC are most associated with high-precision engineering tasks. Like the approach in [1], an illustrative application domain is specialized wirebonder machines used for manufacturing semiconductor components and computer chips [14].

An abstracted model of the motion platforms of such a machine is shown in Fig. 3. The machine base mass is denoted as m_b, and its connection to the ground is modeled with a stiffness coefficient k_b and a viscous damping coefficient d_b. The mass of the moving part along the x-axis, denoted as m_x, is actuated by a control force F_x. A displacement sensor measures the relative motion between m_x and the base m_b (i.e. $x_{21} = x_2 - x_1$, $x = x_{21}$). The relative motion is also influenced by the friction coefficient d_{fric_x}, as well as by the control force F_x.

Namely, the force F_x serves as the control input that actuates the motion of the point of interest (POI) in the x-direction, represented by the mass m_{yz}. The POI (point of interest), represented by the mass m_{yz}, can move in a controlled manner relative to the m_x platform in the y- and z- directions, as well as in the x-direction due to a flexible coupling between m_x and m_{yz}. This coupling is characterized by stiffness k_{xy} and viscous damping d_{xy}. The control objective is to accurately position POI (mass m_{yz}) relative to the base (mass m_b), i.e., to control the displacement $x_{31} = x_3 - x_1$.

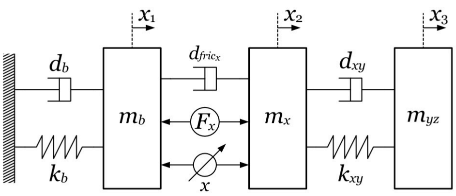

Fig. 3. Abstracted model of the motion platform of a wirebonder machine

Fig. 4 and Fig. 5 illustrate the methodology for generating and applying the ILC signal in a specific application case. In the operational phase of the system, a feedback sensor is available to measure the displacement $x = x_{21} = x_2 - x_1$, along with a pre-generated ILC signal tailored to a given reference trajectory and defined operating conditions, including disturbances and initial values. During the ILC signal generation phase, the system must also be equipped with sensors at the POI to measure the displacement $x_{31} = x_3 - x_1$. In typical operation, the system does not include sensors at the POI.

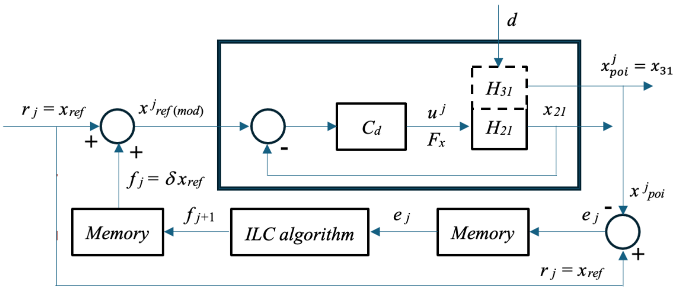

Fig. 4. Serial integration of the ILC signal f_j into the feedback control system with the plant from Fig. 3.

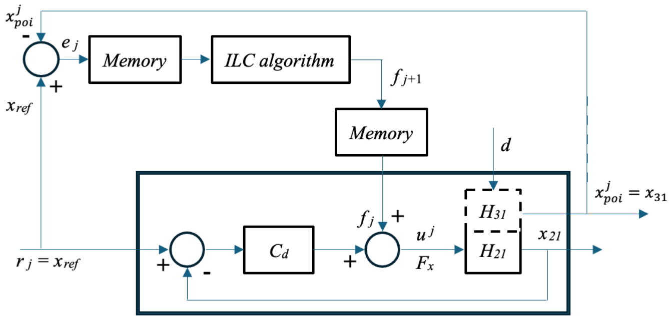

Fig. 5. Parallel integration of the ILC signal f_j into the feedback control system with the plant from Fig. 3.

TABLE I.

Chosen parameters for the model from Fig. 3	
Parameter	***Parameter value***
m_b	225 [kg]
m_x	1.8 [kg]
m_{yz}	2.8 [kg]
k_b	3.3e07 [N/m]
k_{xy}	4.3e07 [N/m]
d_b	3.3e04 [Ns/m]
d_{xy}	640 [Ns/m]
d_{fric_x}	100 [Ns/m]

a. The parameters are not associated with any specific wirebonder machine model.

$$S(z) = \frac{e}{r} = \frac{u}{d} = \frac{1 + C_d(z)\big(H_{21}(z) - H_{31}(z)\big)}{1 + C_d(z)H_{21}(z)}$$
$$PS(z) = S(z)H_{31}(z) \tag{8}$$
$$T_{cs}(z) = \frac{x_{31}}{r} = \big(1 + C_d(z)H_{21}(z)\big)^{-1} C_d(z)H_{31}(z)$$

In accordance with the transformation of the control structures from Fig. 1 and Fig. 2 into the structures shown in Fig. 5 and Fig. 4, the expressions in (2) for the sensitivity function, process sensitivity function, and complementary sensitivity function are transformed into the expressions in (8). For the selected sampling frequency of 10 kHz and a closed-loop system bandwidth of 230 Hz, a feedback controller was designed using frequency shaping methods. Subsequently, the learning filter $L(z)$ and the robustness filter $Q(z)$ were designed according to relations (7) and (6), respectively.

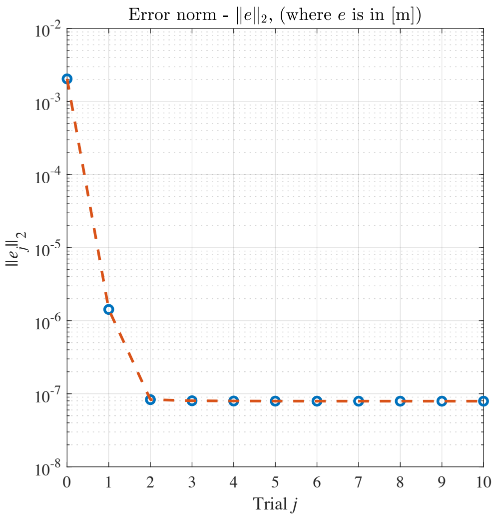

Fig. 6. Demonstration of the impact of ILC signal convergence on x-coordinate tracking accuracy in the control structure shown in Fig. 4 (serial ILC implementation)

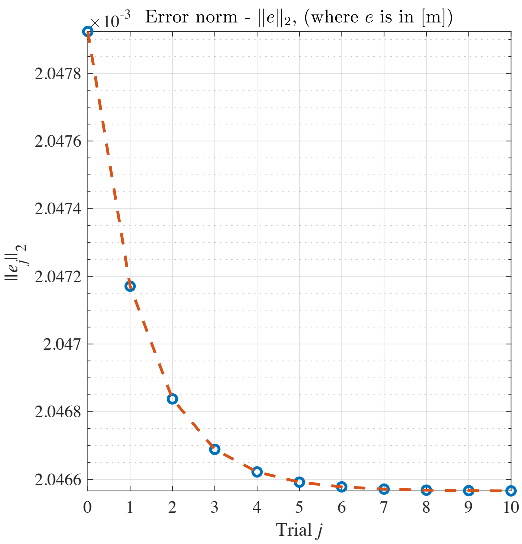

Fig. 7. Demonstration of the impact of ILC signal convergence on x-coordinate tracking accuracy in the control structure shown in Fig. 5 (parallel ILC implementation)

The simulation results presented in Fig. 6 through Fig. 9 demonstrate the superiority of the serial ILC implementation over the parallel one. Figures 6 and 7 illustrate the performance improvements achieved by incorporating the ILC signal into the closed-loop control system (the j = 0 corresponds to system performance with feedback control only, without ILC signal implementation). It is evident that the serial ILC implementation significantly enhances the system's performance for the given reference trajectory. Fig. 9 further shows that the control variable exhibits improved dynamic behavior in the case of serial ILC implementation.

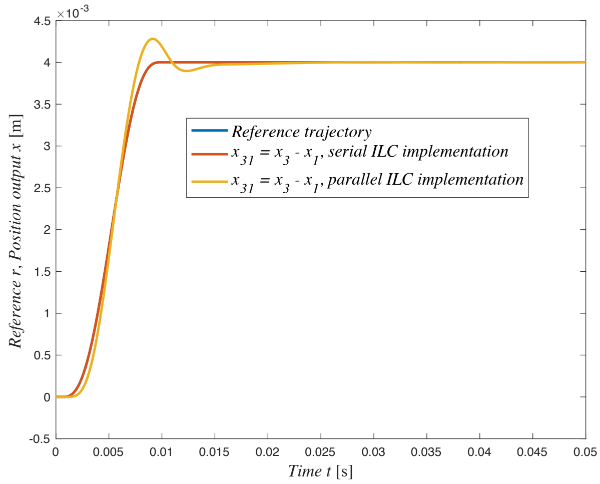

Fig. 8. Desired and actual position output $x_{31} = x_3 - x_1$: Serial ILC shows close tracking of the reference, whereas parallel ILC results in a visible deviation (The blue line indicating the reference trajectory is completely overlapped by the red line representing the achieved output x_{31} which reflects the high tracking accuracy obtained using the serial ILC implementation.)

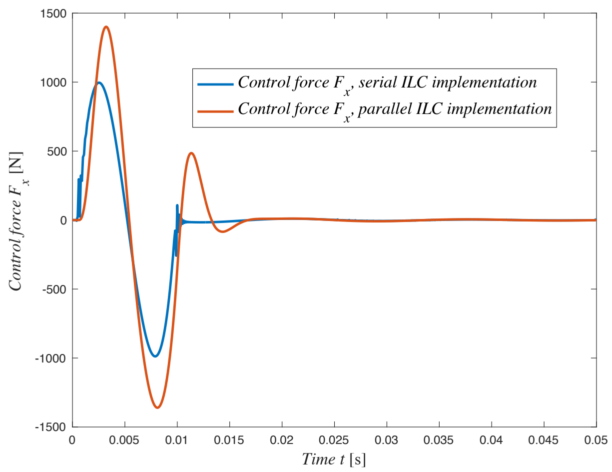

Fig. 9. Control variable (F_x [N]) for serial and parallel ILC implementation

IV. Conclusions

This paper presents the concept of iterative learning of the compensation component of the control signal (ILC signal). The learning phase, which requires sensors at the points of interest (POIs), is distinct from the operational phase, in which only the generated ILC signal is applied (without sensors at the POIs). The structural and parametric synthesis of both parallel and serial integrations of the ILC signal into an existing system is described.

To date, the literature has not emphasized the superiority of either approach to ILC signal synthesis and implementation. However, this study demonstrates through simulation that serial ILC implementation provides superior performance within a closed-loop control system. It results in improved control signal dynamics and more accurate reference trajectory tracking.

Future research should focus on analytically interpreting and further discussing the simulation results and the conclusions drawn from this work. Future work may include experimental validation, as this would provide stronger support than simulations alone.

ACKNOWLEDGMENT

This research has been supported by the research grants of the Serbian Ministry of Science, Technological Development, and Innovations, Grant No. 451-03-137/2025-03/200105 from 04.02.2025. and Grant No. 451-03-136/2025-03/200066.

REFERENCES

[1] M. Matijević and D. Kostić, "Iterative Learning Control – A review of theoretical approaches and industrial applications", Annual Conference of the Scientific Society of Serbia – GODKON 2024, October 31 – November 1, 2024, Belgrade.

[2] H.S. Ahn, Y. Chen, K.L. Moore, "Iterative learning control: Brief survey and categorization", IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), Vol. 37, No. 6, pp. 1099-1121, 2007.

[3] D.A Bristow, M. Tharayil, A.G. Alleyne, "A survey of iterative learning control: A learning-based method for high-performance tracking control", IEEE Contr. Syst. Mag. Vol. 26, pp. 96-114, 2006.

[4] F. Boeren, A. Bareja, T. Kok, et al., "Frequency-domain ILC approach for repeating and varying tasks: With application to semiconductor bonding equipment", IEEE Transaction on Mechatronics 21, 5, pp. 2716-2727, 2016.

[5] K. L. Moore, Iterative Learning Control for Deterministic Systems, Springer-Verlag, London, 1993.

[6] E. Rogers, B. Chu, C. Freeman and P. Lewin, Iterative Learning Control Algorithms and Experimental Benchmarking, Wiley, New York, 2023.

[7] Y. Wang, F. Gao and F. J. Doyle FJ, "Survey on iterative learning control, repetitive control, and run-to-run control", Journal of Process Control, Vol. 19, pp 1589–1600, 2009.

[8] D. Shen and X. Li, "Iterative Learning Control for Systems with Iteration Varying Trial Lengths: Synthesis and Analysis", Springer, Berlin, 2019.

[9] D. Shen and Y. Wang, "Survey on stochastic iterative learning control". Journal of Process Control, Vol. 24, No. 12, pp 64–77, 2014.

[10] D. Shen, Iterative learning control with incomplete information: a survey. IEEE/CAA Journal of Automatica Sinica, Vol. 5, No. 5, pp. 885-901, 2018.

[11] M. Norrlöf, Iterative Learning Control – Analysis, Design, and Experiments, Ph. D. thesis, Linkoping University, Linkoping, Sweden, 2000.

[12] J.J. Bolder, Flexibility and Robustness in Iterative Learning Control: with Applications to Industrial Printers, Ph. D. thesis, TU/e, Eindhoven, Netherlands, 2015.

[13] O. Koçan, Feedback via Iterative Learning Control for Repetitive Systems, Ph. D. thesis, ISAE-SUPAERO, Toulouse, France, 2020.

[14] AERO Bonder, https://semi.asmpt.com/en/products/icd/wb/ball-bonding/eagle-aero/, 2024.

[15] J. van Zundert and T. Oomen, "On inversion-based approaches for feedforward and ILC", Mechatronics, Vol. 50, pp. 282–291, 2018.

[16] M. Matijević, V. Filipović, and D. Kostić, Iterative learning control (ILC) in manufacturing systems: Design of ILC algorithms and overview of model inversion techniques for ILC synthesis, Proc. of 10th International Congress Motor Vehicles & Motors 2024 (ISBN 978-86-6335-120-2), pp 213-227, 2024.

[17] M. Tomizuka, "Zero phase error tracking algorithm for digital control", J. Dyn. Syst. Meas. Control, Vol. 109, No. 1, pp. 65–68, 1987.