PI-Based Set-Point Learning Control for Batch Processes with Unknown Dynamics and Nonrepetitive Uncertainties

Shoulin Hao*, Tao Liu†, Eric Rogers‡, Youqing Wang§, Wojciech Paszke¶

18.11.2024

Abstract

For industrial batch processes with unknown dynamics subject to nonrepetitive initial conditions and disturbances, this paper proposes a novel adaptive data-driven set-point learning control (ADDSPLC) scheme based on only the measured process input and output data, which has two loops, one each for the dynamics within a batch and the other for the batch-to-batch dynamics. In the former case, a model-free tuning strategy is firstly presented for determining the closed-loop PI controller parameters. For the latter case, a set-point learning control law with adaptive set-point learning gain and gradient estimation is developed for batch run optimization. Robust convergence of the output tracking error is rigorously analyzed together with the boundedness of adaptive learning gain and real-time updated set-point command. Moreover, another iterative extended state observer based ADDSPLC scheme is developed with rigorous convergence and boundedness analysis, to enhance the robust tracking performance against nonrepetitive uncertainties. Finally, two illustrative examples from the literature are used to demonstrate the effectiveness and superiority of the new schemes over the recently developed data-driven learning control designs.

Batch processes with unknown dynamics, data-driven control, PI controller tuning, set-point learning control, robust convergence analysis, iterative extended state observer.

1 Introduction

Owing to high flexibility and versatility for functional implementation and product manufacturing, batch processes with repetitive dynamics and operations have been widely constructed in engineering practice, e.g., polymer injection molding [1], pharmaceutical crystallization [2], etc. Over the

*This work is supported in part by the NSF China Grants 61903060, and 62173058, the Fundamental Research Funds for the Central Universities of China (DUT22RC(3)020) and National Science Centre in Poland, grant No. 2020/39/B/ST7/01487. The corresponding authors are Tao Liu and Eric Rogers.

†Shoulin Hao and Tao Liu are with the Key Laboratory of Intelligent Control and Optimization for Industrial Equipment of Ministry of Eduction, and the Institute of Advanced Control Technology, Dalian University of Technology, Dalian, 116024, China (e-mail: slhao@dlut.edu.cn; liurouter@ieee.org).

‡Eric Rogers is with the School of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, UK (e-mail: etar@ecs.soton.ac.uk)

§Yongqing Wang is with the College of Information Science and Technology, Beijing University of Chemical Technology, Beijing 100029, China (e-mail: wang.youqing@ieee.org)

¶Wojciech Paszke is with the Institute of Automation, Electronic and Electrical Engineering, University of Zielona Góra, ul.Szafrana 2, 65-246 Zielona Góra, Poland (e-mail: w.paszke@iee.uz.zgora.pl)

past decades, a large amount of research efforts has been devoted to advanced control methods and technologies for various batch processes in industry. Among them, iterative learning control (ILC) has attracted broad attention due to its capability of gradually improving system performance by making use of the historical batch run data with no or moderate process model information, as surveyed in [3, 4, 5].

Since the pioneering work in [6], a lot of ILC methods have been studied for linear/nonlinear batch processes and repetitive systems operated in a finite duration, based on the lifting technique [7], two-dimensional (2D) system theory [8, 9], contraction mapping principle [10], norm-optimal technique [11], etc. Recently, it has been clarified that the conventional ILC, also named as open-loop ILC, is able to maintain robust convergence of the resulting control system in the presence of nonrepetitive process uncertainties, initial conditions and external disturbances [12, 13]. However, the conventional ILC method likely leads to unacceptable tracking errors in the initial batch runs, especially for open-loop unstable processes, even though the batch-direction convergence of tracking errors could be guaranteed. In turn, the current iteration tracking error (CITE) was considered in the design of ILC updating law in [14] to expedite the convergence speed by increasing the learning gain associated with the CITE. Subsequently, real-time process state and output information were also taken into account in the ILC design through an identical closed-loop controller, forming the so-called direct-type ILC framework (see e.g., [15, 16]).

In contrast to the direct-type ILC methods, indirect-type ILC has gained increasing attention in recent years owing to that real-time feedback control and feedforward learning control could be separately designed. In particular, if the feedback control structure is configured a priori for the controlled process, it suffices to design a set-point learning controller for realizing batch optimization without modifying the existing closed-loop structure, which is more appealing in various engineering applications. By using the 2D Roesser system description, a PID-type indirect ILC was proposed in [17] for batch processes with time-varying uncertainties, followed by an improved PI-based indirect-type ILC method [18] designed in terms of the 2D Fornasini-Marchesini model framework. In [19], a set-point related indirect-type ILC design was presented based on a model predictive control scheme to address the process output constraints. Different from the previous indirect-type ILC methods based on a unity feedback control structure, the recent paper [20] developed an extended state observer (ESO) based indirect-type ILC for batch processes with time- and batch-varying uncertainties, such that the set-point tracking and disturbance rejection performance could be evidently enhanced owing to the intrinsic two-degree-of-freedom tuning properties of the inner-loop feedback control structure. Nevertheless, most of the existing indirect-type ILC designs, together with the corresponding convergence analysis, basically depend on the process modeling or dynamics information, thus limiting their applications to complex industrial processes that are difficult to model.

With the increase of scale and complexity for modern industry and manufacturing systems, it becomes more time-consuming or even impossible to model complex process dynamics accurately by using the traditional first-principle or system identification methods. Data-driven control (DDC), using only the measured input/output (I/O) data of the controlled processes, has become a very active research area, see, e.g., the monographs [21, 22], survey papers [23, 24] and the references therein. For example, Meng et al. [25] proposed an optimization-based design for data-driven learning control of nonlinear systems subject to time-varying dynamics, along with convergence analysis. By using a radial basis function neural network to estimate the unknown pseudo-partial derivative of a dynamic linearization model and nonrepetitive external disturbances, a data-driven predictive ILC scheme was developed in [26] for repeatable nonaffine nonlinear discrete-time systems subject to nonrepetitive disturbances. Recently, a controller-dynamic-linearization-based data-driven ILC (DDILC) was designed in [27] for unknown nonlinear systems, where the dynamic linearization technique was applied to not only the controlled nonlinear system but also the nonlinear learning controller. To improve the system tracking performance, a data-driven high-order optimal ILC was developed in [28] for a class of nonlinear repetitive control systems, by introducing the tracking errors and control inputs from previous iterations into the ILC updating law.

Note that the above-mentioned DDILC methods directly update the control input and therefore, could not guarantee the convergence or stability of the resulting learning system when applied to a batch process with an a priori feedback control loop. To facilitate the application to linear/nonlinear batch processes with an inherent P-type feedback control structure, an indirect adaptive DDILC method was recently proposed in [29] to improve the system tracking performance by updating the set-point command for batch run. In fact, the proportional-integral (PI) controller has much wider applications in engineering practice [30] because it can eliminate the steady-state tracking error and counteract a constant-type disturbance, compared with the P-type feedback controller. Recently, it has been revealed that the PI/PID controller could be used to stabilize uncertain nonlinear systems [31, 32, 33]. However, it remains open as yet to tune a PI or PID controller by only using the system I/O data without any prior knowledge of the system dynamics, let alone the combination with a data driven set-point learning control (DDSPLC) scheme for batch run optimization under unknown system/batch dynamics and nonrepetitive uncertainties. These issues motivate this study.

In this paper, a robust PI-based adaptive DDSPLC (PI-ADDSPLC) scheme is proposed to realize performance optimization of batch processes with unknown dynamics and an inherent PI feedback control structure, based on only the measured process I/O data rather than an a priori model. A model-free tuning strategy of PI controller is firstly presented to determine a set of time-invariant PI controller parameters to facilitate the initial batch run and the subsequent set-point learning control design. Then an ADDSPLC is developed to regulate the set-point command of the established closed-loop PI control structure for realizing batch run optimization. The robust convergence of tracking error along with the boundedness of adaptive learning gain and set-point command is analyzed by the mathematical induction. Moreover, to enhance the tracking performance in the presence of nonrepetitive uncertainties, an iterative ESO (IESO) based PI-ADDSPLC is further designed, along with robust convergence and boundedness analysis. Two illustrative examples from the literature are adopted to validate the effectiveness and advantage of the proposed schemes. The main contributions of this paper include:

(i) A novel ADDSPLC scheme based on the widely used PI control loop is established for linear/nonlinear batch processes with unknown dynamics subject to nonrepetitive initial conditions and disturbances. The system tracking performance could be significantly improved from the first batch and on, compared to the recently developed indirect-type DDILC methods (e.g., [34] based on a P-type feedback control loop) and direct-type DDILC methods (e.g., [25]). Moreover, a feasible data-driven PI tuning algorithm is provided to facilitate setting up a PI control loop for the set-point learning control design without any priori knowledge of the process dynamics.

(ii) To proactively suppress nonrepetitive uncertainties such as initial condition shifts and batch-varying disturbances, another IESO based PI-ADDSPLC scheme is proposed by introducing an IESO into the PI-ADDSPLC scheme to estimate nonrepetitive uncertainties for counteraction, such that the output tracking accuracy is further improved. The new scheme substantially extends the time-domain ESO-based data-driven control method recently developed in [34] to an iterative-domain design of DDILC for batch run optimization.

(iii) Robust convergence of the output tracking error along with the boundedness of adaptive learning gain and set-point command is rigorously analyzed by the mathematical induction for both of the new designs.

For clarity, the remainder of this paper is structured as follows. Section 2 briefly introduces the research problem and required preliminaries. Next, a model-free tuning strategy of PI controller parameters is given in Section 3. The proposed robust PI-ADDSPLC scheme is detailed in Section 4, followed by robust convergence and boundedness analysis for the resulting ILC system in Section 5. In Section 6, another IESO based PI-ADDSPLC scheme is developed, together with the corresponding robust convergence analysis. Two examples from the recent literature are used in Section 7 to demonstrate the effectiveness and superiority of the new schemes. Finally, Section 8 summarizes the main developments in this paper and discusses possible future research.

Notation: Throughout the paper, the following notation is used. $\mathbb{Z}_+ = \{0, 1, \ldots\}$, $\mathbb{Z}_N = \{0, 1, \ldots, N\}$ and $\mathbb{Z}_N/\{0\} = \{1, \ldots, N\}$ for any $N \in \mathbb{Z}_+$. $\mathbb{R}^n$ and $\mathbb{R}^{n\times m}$ denote n-dimensional Euclidean space and $n \times m$ real matrix spaces, respectively. I and 0, respectively, denote the identity and null matrices with compatible dimensions. Also, $|\cdot|$ and $\|\cdot\|$ denote the absolute value and the Euclidean norm of the argument, respectively. For a matrix A, $A^\top$ denotes its transpose, and for any function $f_k(t)$ where k and t denote the batch and time indices, respectively, denote by $\Delta f_k(t) = f_k(t) - f_{k-1}(t)$ and $\Delta_t f_k(t) = f_k(t) - f_k(t-1)$ the difference functions along the batch and time directions, respectively.

2 Problem formulation and preliminaries

Consider a batch process with unknown dynamics described by

$$\begin{aligned} y_k(t+1) = f\big(&y_k(t), y_k(t-1), \ldots, y_k(t-n_y),\\ &u_k(t), u_k(t-1), \ldots, u_k(t-n_u)\big) + \omega_k(t), \end{aligned} \tag{1}$$

where $t \in \mathbb{Z}_{N-1}$ and $k \in \mathbb{Z}_+$ are, respectively, the time step and batch number; N is the total length of each batch; $y_k(t) \in \mathbb{R}$ and $u_k(t) \in \mathbb{R}$ are the process output and input at time t of batch k, respectively; $\omega_k(t)$ denotes the nonrepetitive disturbances acting on the process dynamics, and includes batch-to-batch variation of the process dynamics; $f(\cdot)$ is an unknown linear/nonlinear function with unknown input order n_u and output order n_y. The initial condition in (1) is taken as $y_k(0) = y_0 + \delta_k$, where y_0 is an identical initial resetting for a batch run and δ_k denotes nonrepetitive initial shifts. Hereafter, the unknown function $f(\cdot)$ in (1) is rewritten as $f(x_1, x_2, \ldots, x_{n_u+n_y+2})$ for the notational brevity, where $x_i \in \mathbb{R}$, $i = 1, 2, \ldots, n_u + n_y + 2$, denotes the i-th variable of $f(\cdot)$.

The objective in this paper is to design a robust adaptive set-point learning control scheme based on only the measured I/O data for a batch process described by (1) with a PI feedback control structure, such that the process output tracks the desired reference trajectory as closely as possible in the presence of nonrepetitive external disturbances and initial shifts along the batch direction, i.e., for any $t \in \mathbb{Z}_N/\{0\}$

$$\sup_{k\in\mathbb{Z}_+} |e_k(t)| \leq \beta_e,\ \limsup_{k\to\infty} |e_k(t)| \leq \beta_{e_{\sup}}, \tag{2}$$

where $e_k(t) \triangleq y_{\rm d}(t) - y_k(t)$, $y_{\rm d}(t)$ denotes the desired reference trajectory and satisfies $|y_{\rm d}(t)| \leq \beta_{\rm d} < \infty$, where $\beta_{\rm d}$ is a finite constant, β_e and $\beta_{e_{\sup}}$ satisfying $\beta_e > \beta_{e_{\sup}} \geq 0$ are related to the upper bounds of nonrepetitive external disturbance variation and initial shifts. Moreover, the set-point command $y_k^s(t)$ is required to be bounded for implementation, i.e.

$$\sup_{k\in\mathbb{Z}_+} \max_{t\in\mathbb{Z}_{N-1}} |y_k^s(t)| \leq \beta_s < \infty, \tag{3}$$

where $\beta_s > 0$ is a finite constant.

For the simplicity of analysis, the following assumptions are made along with a technical lemma.

Assumption 1 $u_0(t) = 0$ *for any* $t \in \mathbb{Z}_{N-1}$ *and* $y_k(t) = 0$ *if* $t < 0$ *or* $k < 0$.

Assumption 2 *[25] The function f is continuously differentiable such that the partial derivatives with respect to its arguments are bounded, i.e.*

$$\left|\frac{\partial f}{\partial x_i}(x_1, x_2, \ldots, x_{n_y+n_u+2})\right| \leq \beta_{\overline{f}},$$
$$\forall x_i \in \mathbb{R}, i = 1, 2, \ldots, n_y + n_u + 2, \forall t \in \mathbb{Z}_{N-1},$$

where $\beta_{\overline{f}} > 0$ is a finite constant. Furthermore, let $\partial f/\partial x_{n_y+2}$ be sign-fixed, which is assumed to be positive, without loss of generality.

Assumption 3 *[21, 36] A batch process described by* (1) *satisfies the following generalized Lipschitz condition along the time direction*

$$|\Delta_t y_k(t+1)| \leq \beta_\Gamma \|\Delta_t H_k(t)\| \tag{4}$$

for $\|\Delta_t H_k(t)\| \neq 0$, where $\beta_\Gamma > 0$ is a finite constant and $\Delta_t H_k(t) = [\Delta_t y_k(t)\ \Delta_t u_k(t)]^\top$.

Assumption 4 *[35, 20] The nonrepetitive external disturbances and initial condition shifts are bounded, i.e.*

$$|\omega_k(t)| \leq \beta_\omega,\ \forall t \in \mathbb{Z}_{N-1},\ \forall k \in \mathbb{Z}_+,$$
$$|\delta_k| \leq \beta_\delta,\ \forall k \in \mathbb{Z}_+,$$

where β_ω and β_δ are finite positive constants.

Assumption 1 reflects the widely used initial zero or steady state of the process dynamics for the convenience of control design and stability analysis. Assumptions 2 and 3 are widely used in the research fields of nonlinear system control and data-driven control (e.g., [21, 22, 23, 24, 31, 25, 32, 33, 36]) to characterize the process nonlinearity, i.e., the process output change is constrained by the changes in all the input and past output variables. Assumption 4 is also widely used for ILC design of batch/repetitive processes with nonrepetitive uncertainties in the literature (e.g., [12, 35]), owing to physical constraints of batch process operation in practice.

Lemma 1 *[35] Under Assumption 2, the batch process description in* (1) *can be equivalently transformed into the following extended dynamic linearization data model (DLDM)*

$$\begin{aligned}\boldsymbol{y}_i - \boldsymbol{y}_j = {} & \boldsymbol{\Phi}_{i,j}(\boldsymbol{u}_i - \boldsymbol{u}_j) + \boldsymbol{\Upsilon}_{i,j}(\boldsymbol{\omega}_i - \boldsymbol{\omega}_j) \\ & + \boldsymbol{\vartheta}_{i,j}(\delta_i - \delta_j), \quad \forall i, j \in \mathbb{Z}_+,\end{aligned} \tag{5}$$

where

$$\begin{aligned}\boldsymbol{\Phi}_{i,j} &= \begin{bmatrix} \phi^0_{i,j}(0) & 0 & \cdots & 0 \\ \phi^1_{i,j}(0) & \phi^1_{i,j}(1) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \phi^{N-1}_{i,j}(0) & \phi^{N-1}_{i,j}(1) & \cdots & \phi^{N-1}_{i,j}(N-1) \end{bmatrix}, \\ \boldsymbol{\Upsilon}_{i,j} &= \begin{bmatrix} 1 & 0 & \cdots & 0 \\ \upsilon^1_{i,j}(0) & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \upsilon^{N-1}_{i,j}(0) & \upsilon^{N-1}_{i,j}(1) & \cdots & 1 \end{bmatrix}, \\ \boldsymbol{\vartheta}_{i,j} &= \begin{bmatrix} \vartheta^0_{i,j} & \vartheta^1_{i,j} & \cdots & \vartheta^{N-1}_{i,j} \end{bmatrix}^\top, \\ \boldsymbol{y}_i &= \begin{bmatrix} y_i(1) & y_i(2) & \cdots & y_i(N) \end{bmatrix}^\top, \\ \boldsymbol{u}_i &= \begin{bmatrix} u_i(0) & u_i(1) & \cdots & u_i(N-1) \end{bmatrix}^\top, \\ \boldsymbol{\omega}_i &= \begin{bmatrix} \omega_i(0) & \omega_i(1) & \cdots & \omega_i(N-1) \end{bmatrix}^\top \end{aligned}$$

and all nonzero elements of $\boldsymbol{\Phi}_{i,j}$, $\boldsymbol{\Upsilon}_{i,j}$ *and* $\boldsymbol{\vartheta}_{i,j}$ *are bounded, i.e., there exists a finite bound* $\beta_\phi > 0$ *such that*

$$\begin{aligned} |\phi^t_{i,j}(\xi)| &\le \beta_\phi, \ \forall \xi \in \mathbb{Z}_t, \forall t \in \mathbb{Z}_{N-1}, \forall i, j \in \mathbb{Z}_+, \\ |\upsilon^t_{i,j}(\xi)| &\le \beta_\phi, \ \forall \xi \in \mathbb{Z}_t, \ \forall t \in \mathbb{Z}_{N-1}, \ \forall i, j \in \mathbb{Z}_+, \\ |\vartheta^t_{i,j}| &\le \beta_\phi, \ \forall t \in \mathbb{Z}_{N-1}, \ \forall i, j \in \mathbb{Z}_+. \end{aligned} \tag{6}$$

In contrast to the standard DLDM that characterizes the iterative dynamic relationship between the process output and input at two consecutive batches, the extended DLDM further generalizes the relationship to any two batches. Moreover, the parameter matrix $\boldsymbol{\Phi}_{i,j}$ and uncertainty information composed of external disturbances and initial shifts could be separately estimated in the extended DLDM.

Taking $i = k$ and $j = k - 1$ in (5) gives

$$\Delta y_k(t+1) = \boldsymbol{\phi}^\top_{k,k-1}(t)\Delta\boldsymbol{u}_k(t) + \chi_k(t), \tag{7}$$

where $\chi_k(t) = \Delta\boldsymbol{\omega}^\top_k(t)\boldsymbol{\upsilon}_{k,k-1}(t) + \vartheta^t_{k,k-1}\Delta\delta_k$ and

$$\begin{aligned} \boldsymbol{\phi}_{k,k-1}(t) &= \begin{bmatrix} \phi^t_{k,k-1}(0) & \phi^t_{k,k-1}(1) & \cdots & \phi^t_{k,k-1}(t) \end{bmatrix}^\top, \\ \Delta\boldsymbol{u}_k(t) &= \begin{bmatrix} \Delta u_k(0) & \Delta u_k(1) & \cdots & \Delta u_k(t) \end{bmatrix}^\top, \\ \Delta\boldsymbol{\omega}_k(t) &= \begin{bmatrix} \Delta\omega_k(0) & \Delta\omega_k(1) & \cdots & \Delta\omega_k(t) \end{bmatrix}^\top, \\ \boldsymbol{\upsilon}_{k,k-1}(t) &= \begin{bmatrix} \upsilon^t_{k,k-1}(0) & \upsilon^t_{k,k-1}(1) & \cdots & 1 \end{bmatrix}^\top. \end{aligned}$$

Using Assumption 4 and Lemma 1, it follows that $\chi_k(t)$ is bounded for any $t \in \mathbb{Z}_{N-1}$ and $k \in \mathbb{Z}_+$ and assumed to satisfy $\sup_{t\in\mathbb{Z}_{N-1},k\in\mathbb{Z}_+} |\chi_k(t)| \leq \beta_\chi(\beta_{\Delta\omega}, \beta_{\Delta\delta})$, where $\beta_\chi(\beta_{\Delta\omega}, \beta_{\Delta\delta}) > 0$ is a finite constant dependent on the finite bounds of $\sup_{t\in\mathbb{Z}_{N-1},k\in\mathbb{Z}_+} |\Delta\omega_k(t)|$ and $\sup_{k\in\mathbb{Z}_+} |\Delta\delta_k|$ defined by $\beta_{\Delta\omega}$ and $\beta_{\Delta\delta}$, respectively. For the notational brevity, a symbol "•" is used to indicate the relevance to $\beta_{\Delta\omega}$ and $\beta_{\Delta\delta}$ in the later sections.

3 Data-driven PI controller tuning

A data-driven PI tuning method is developed in this section to set up the inherent PI feedback control loop. For this purpose, the following time-domain DLDM is established for a batch process described by (1) under Assumptions 2 and 3

$$\Delta_t y(t+1) = \Gamma(t)\Delta_t H(t), \tag{8}$$

where $\Gamma(t) = [\gamma_1(t)\ \gamma_2(t)]$ is a time-varying pseudo-gradient vector dependent on the process input and output in the time domain. More details on the model transformation (8) are given in [21]. In this section, the batch index k is omitted since it is irrelevant to the time-domain PI controller tuning.

To tune the PI controller in the time domain, the following performance function is introduced

$$J_1(u(t)) = |y_\mathrm{d}(t+1) - y(t+1)|^2 + \lambda_t |u(t) - u(t-1)|^2,$$

where $\lambda_t > 0$ is a user-specified weighting factor for evaluating the impact of input variation in the time domain. Taking the first-order derivative of $J_1(u(t))$ with respect to $u(t)$ and equating the result to zero give

$$\Delta_t u(t) = \frac{\rho_t \gamma_2(t)}{\lambda_t + \gamma_2^2(t)} \left[y_\mathrm{d}(t+1) - y(t) - \gamma_1(t)\Delta_t y(t)\right], \tag{9}$$

where $\rho_t \in (0, 2)$ is a tuning parameter to offer a flexible feedback control law. Under a constant set-point command, the increment of control input in (9) could be reformulated as

$$\begin{aligned} \Delta_t u(t) &= \frac{\rho_t \gamma_2(t)}{\lambda_t + \gamma_2^2(t)} \left[e(t) + \gamma_1(t)\Delta_t e(t)\right] \\ &= \tau_\mathrm{P}(t)\Delta_t e(t) + \tau_\mathrm{I}(t)e(t), \end{aligned} \tag{10}$$

where

$$\tau_\mathrm{P}(t) \triangleq \frac{\rho_t \gamma_1(t)\gamma_2(t)}{\lambda_t + \gamma_2^2(t)},\ \tau_\mathrm{I}(t) \triangleq \frac{\rho_t \gamma_2(t)}{\lambda_t + \gamma_2^2(t)}.$$

It is seen that the time-domain increment of the control input in (10) is exactly the PI controller in a difference equation form. Note that in the presence of a time-varying set-point command, the closed-loop stability under the PI controller in (10) could be maintained for a finite variation of the set-point command, e.g., a ramp type or sine signal with low frequency, see [17, 18] for more details.

To estimate the unknown $\gamma_1(t), \gamma_2(t)$ for determining the feasible tuning regions of the PI controller parameters, the following parameter estimation algorithm [36] is used

$$\begin{aligned}\hat{\Gamma}(t) &= \hat{\Gamma}(t-1) + \frac{\eta_t \Delta_t H(t-1)}{\mu_t + \|\Delta_t H(t-1)\|^2} \\ &\quad \times \left[\Delta_t y(t) - \hat{\Gamma}^\top(t-1)\Delta_t H(t-1)\right], \\ \hat{\Gamma}(t) &= \hat{\Gamma}(0), \text{ if } \|\hat{\Gamma}(t)\| \le \varepsilon_t, \text{ or } \|\Delta_t H(t-1)\| \le \varepsilon_t, \\ &\quad \text{or } \operatorname{sign}(\hat{\Gamma}(t)) \ne \operatorname{sign}(\hat{\Gamma}(0)),\end{aligned} \tag{11}$$

where $\eta_t \in (0,2)$ is a tuning parameter, $\mu_t > 0$ is a weighting factor, $\hat{\Gamma}(t) = [\hat{\gamma}_1(t)\ \hat{\gamma}_2(t)]$, $\hat{\gamma}_1(t)$ and $\hat{\gamma}_2(t)$ are the respective estimates of $\Gamma(t)$, $\gamma_1(t)$ and $\gamma_2(t)$, and ε_t is a sufficiently small constant for initialization. The boundedness of $\hat{\Gamma}(t)$ and the time-domain convergence of the closed-loop system could be analyzed following the similar way as that in [21] and [36], respectively.

Based on the estimated $\hat{\gamma}_1(t)$ and $\hat{\gamma}_2(t)$, the feasible tuning regions of PI controller parameters can be estimated as

$$\tau_{\mathrm{P}} \in \left[\tau_{\mathrm{P}}^{\min}, \tau_{\mathrm{P}}^{\max}\right], \ \tau_{\mathrm{I}} \in \left[\tau_{\mathrm{I}}^{\min}, \tau_{\mathrm{I}}^{\max}\right], \tag{12}$$

where

$$\begin{aligned}\tau_{\mathrm{P}}^{\min} &= \min_{t\in\mathbb{Z}_{N-1}} \frac{\rho_t \hat{\gamma}_1(t)\hat{\gamma}_2(t)}{\lambda_t + \hat{\gamma}_2^2(t)}, \ \tau_{\mathrm{P}}^{\max} = \max_{t\in\mathbb{Z}_{N-1}} \frac{\rho_t \hat{\gamma}_1(t)\hat{\gamma}_2(t)}{\lambda_t + \hat{\gamma}_2^2(t)}, \\ \tau_{\mathrm{I}}^{\min} &= \min_{t\in\mathbb{Z}_{N-1}} \frac{\rho_t \hat{\gamma}_2(t)}{\lambda_t + \hat{\gamma}_2^2(t)}, \ \tau_{\mathrm{I}}^{\max} = \max_{t\in\mathbb{Z}_{N-1}} \frac{\rho_t \hat{\gamma}_2(t)}{\lambda_t + \hat{\gamma}_2^2(t)}.\end{aligned}$$

Given (12), a tuning procedure is presented by Algorithm 1, which can be used to determine the desired time-invariant PI controller parameters.

Algorithm 1 (PI controller tuning)

Input: Initial PI controller parameters $\tau_{\mathrm{P}} = \tau_{\mathrm{P}}^{\min}$ and $\tau_{\mathrm{I}} = \tau_{\mathrm{I}}^{\min}$, tuning parameters ρ_t and η_t, weighting factors λ_t and μ_t, user specified thresholds ε_t and ϱ, step sizes $\Delta\tau_{\mathrm{P}}$ and $\Delta\tau_{\mathrm{I}}$ for iteration.

Output: Desired PI controller parameters τ_{P} and τ_{I}.

while ($\tau_{\mathrm{P}} \le \tau_{\mathrm{P}}^{\max}$) **do**
  while ($\tau_{\mathrm{I}} \le \tau_{\mathrm{I}}^{\max}$) **do**
    Apply (9) and (11) to a batch process described by (1) and compute the averaged tracking error (ATE) defined by $\text{ATE} = \sum_{t=1}^{N} |e(t)|/N$;
    if (ATE $\le \varrho$) **then**
      Determine the desired τ_{P} and τ_{I};
    else if (ATE $> \varrho$) **then**
      $\tau_{\mathrm{I}} = \tau_{\mathrm{I}} + \Delta\tau_{\mathrm{I}}$;
    end if
  end while
  $\tau_{\mathrm{P}} = \tau_{\mathrm{P}} + \Delta\tau_{\mathrm{P}}$, $\tau_{\mathrm{I}} = \tau_{\mathrm{I}}^{\min}$;
end while

Remark 1 *For unknown linear time-invariant systems with a constant-type set-point command, the PI controller parameters obtained by the above procedure will ultimately converge to steady-state values, which could be taken as the desired PI controller parameters. Note that the PI controller parameters may be time-varying without steady-state values for a nonlinear time-varying system. In such a case, a preferred option is to choose a set of time-invariant PI controller parameters from the estimated tuning regions in* (12) *and fix them for implementation, so as to facilitate the subsequent set-point learning control design.*

Remark 2 *In case no desired PI controller parameters could be obtained for a prescribed ϱ in the above Algorithm 1, one option is to increase ϱ until a set of satisfactory PI controller parameters is obtained. Another option is to assess the ATE index for all computed PI controller parameters within the estimated tuning regions and choose the optimal setting in terms of the minimum of ATE. In addition to the ATE index, other performance indices, e.g., mean-square error and maximum absolute error, may also be used to evaluate the control performance for determining the PI controller parameters. Note that all the corresponding ATE indices in the above tuning regions could be computed offline, and hence the optimal setting of PI controller parameters are available before online implementation.*

Remark 3 *The recently developed data-driven PID controller designs [31, 32, 33] cannot be applied to determine the PI controller parameters or stabilizing regions of the PI controller for a batch process described by* (1) *with unknown dynamics, since the upper bounds of partial derivatives of $f(\cdot)$ with respect to $y(t)$, $y(t-1)$ and $u(t)$ cannot be known in advance. This restriction is removed from the new tuning algorithm.*

4 Robust PI-ADDSPLC scheme

The new robust PI-ADDSPLC scheme is shown in Fig. 1, where the closed-loop PI control structure is within the red dashed-line box, and the rest is the set-point learning control law together with a gradient estimator and another learning gain estimator responsible for regulating the set-point command of the closed-loop control structure from batch to batch, "MEMORY" denotes a storage used to record output tracking error ($e_k(t)$), set-point command ($y^s_k(t)$), process input ($u_k(t)$), process output ($y_k(t)$) and gradient estimation ($\hat{\boldsymbol{\phi}}_{k,k-1}(t)$) in the current batch, and provide the historical batch information (e_{k-1}, y^s_{k-1}, $\Delta \boldsymbol{u}_{k-1}$, Δy_{k-1}, $\hat{\boldsymbol{\phi}}_{k-1,k-2}$) together with the current batch information ($\Delta u_k(t-1)$, $\Delta y_k(t-1)$, $\Delta \boldsymbol{y}^s_k(t-1)$, $\hat{\boldsymbol{\phi}}_{k,k-1}(t)$). Note that the closed-loop PI controller is fixed for each batch run, and therefore independent of the set-point learning control for batch performance optimization. Moreover, only the PI feedback control loop is executed in the initial batch where the set-point command is directly set as the desired reference trajectory $y_{\rm d}$.

Based on the determined PI controller parameters in Section 3, the process input takes the form

$$u_k(t)=\tau_{\rm P} e^s_k(t)+\tau_{\rm I}\sum_{i=0}^{t} e^s_k(i)=\tau e^s_k(t)+\tau_{\rm I}\sum_{i=0}^{t-1} e^s_k(i), \tag{13}$$

where $\tau \triangleq \tau_{\rm P}+\tau_{\rm I}$, $\tau_{\rm P}$ and $\tau_{\rm I}$ are the PI controller parameters, and $e^s_k(t)$ is the set-point tracking error defined by

$$e^s_k(t) = y^s_k(t) - y_k(t), \tag{14}$$

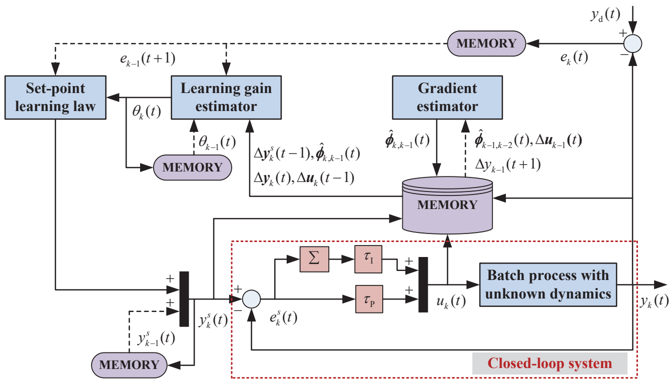

Figure 1: Schematic of the new PI-ADDSPLC design

which differs from the output tracking error $e_k(t)$.

In this paper, an adaptive set-point learning law is designed as follows

$$y_k^s(t) = y_{k-1}^s(t) + \alpha\theta_k(t)e_{k-1}(t+1), \tag{15}$$

where $\theta_k(t)$ is an adaptive set-point learning gain to be determined, and α is a user-specified tuning parameter for regulating the batch-direction convergence speed of the learning system.

To estimate the learning gain $\theta_k(t)$ for control implementation, the following performance evaluation function is used

$$J_2(\theta_k(t)) = |e_k(t+1)|^2 + \lambda |\theta_k(t) - \theta_{k-1}(t)|^2, \tag{16}$$

where $\lambda > 0$ is a user-specified weighting factor to evaluate the impact from variation of $\theta_k(t)$ along the batch direction, and therefore adjust the convergence speed of the parameter estimation algorithm.

It follows from (13) and (14) that

$$\begin{aligned}\Delta u_k(t) &= \tau(\Delta y_k^s(t) - \Delta y_k(t)) + \tau_{\mathrm{I}} \sum_{i=0}^{t-1}(\Delta y_k^s(i) - \Delta y_k(i)) \\ &= \tau\alpha\theta_k(t)e_{k-1}(t+1) - \tau\Delta y_k(t) \\ &\quad + \tau_{\mathrm{I}} \sum_{i=0}^{t-1}(\Delta y_k^s(i) - \Delta y_k(i)).\end{aligned} \tag{17}$$

Then, it follows from the definition of $e_k(t)$ that

$$\begin{aligned}e_k(t+1) &= y_{\mathrm{d}}(t+1) - y_k(t+1) \\ &= e_{k-1}(t+1) - \sum_{j=0}^{t-1} \phi_{k,k-1}^t(j)\Delta u_k(j) - \chi_k(t) \\ &\quad - \phi_{k,k-1}^t(t)\Bigg\{\tau\alpha\theta_k(t)e_{k-1}(t+1) - \tau\Delta y_k(t) \\ &\quad + \tau_{\mathrm{I}} \sum_{i=0}^{t-1}(\Delta y_k^s(i) - \Delta y_k(i))\Bigg\}.\end{aligned} \tag{18}$$

Taking the first-order derivative of $J_2(\theta_k(t))$ with respect to $\theta_k(t)$ and equating the result to zero give

$$\begin{aligned}\theta_k(t) = \theta_{k-1}(t) + \frac{\rho\nu_k(t)}{\lambda+\nu_k^2(t)}\Bigg\{ & e_{k-1}(t+1) - \chi_k(t) \\ & - \theta_{k-1}(t)\nu_k(t) - \tau_{\mathrm{I}}\phi^t_{k,k-1}(t)\sum_{i=0}^{t-1}[\Delta y^s_k(i) - \Delta y_k(i)] \\ & - \sum_{i=0}^{t-1}\phi^t_{k,k-1}(i)\Delta u_k(i) + \tau\phi^t_{k,k-1}(t)\Delta y_k(t)\Bigg\},\end{aligned} \tag{19}$$

where $\nu_k(t) \triangleq \tau\alpha\phi^t_{k,k-1}(t)e_{k-1}(t+1)$, $\rho \in (0,2)$ is a tuning parameter to offer a flexible parameter estimation algorithm.

It is seen from (19) that the increment of $\theta_k(t)$ cannot be realized since $\chi_k(t)$ and $\phi^t_{k,k-1}(i), i = 0,1,\ldots,t$ are unavailable in advance. To address this issue, the uncertainty term $\chi_k(t)$ is omitted and the unknown parameter $\phi^t_{k,k-1}(i)$ is replaced by its estimate obtained by a parameter estimation algorithm [37] along the batch direction, i.e.

$$\begin{aligned}\hat{\boldsymbol{\phi}}_{k,k-1}(t) = \hat{\boldsymbol{\phi}}_{k-1,k-2}(t) + & \frac{\eta\Delta\boldsymbol{u}_{k-1}(t)}{\mu + \|\Delta\boldsymbol{u}_{k-1}(t)\|^2} \\ & \times\left(\Delta y_{k-1}(t+1) - \hat{\boldsymbol{\phi}}^\top_{k-1,k-2}(t)\Delta\boldsymbol{u}_{k-1}(t)\right),\end{aligned} \tag{20}$$

where $\hat{\boldsymbol{\phi}}_{k,k-1}(t) = [\hat{\phi}^t_{k,k-1}(0),\ldots,\hat{\phi}^t_{k,k-1}(t)]$ is the estimation of $\boldsymbol{\phi}_{k,k-1}(t)$, $\mu > 0$ and $\eta \in (0,2)$ are two tuning parameters. Moreover, the boundedness of $\hat{\boldsymbol{\phi}}_{k,k-1}(t)$ was established in [37]. For ease of the subsequent analysis, the upper bound of $\hat{\boldsymbol{\phi}}_{k,k-1}(t)$ is denoted by $\beta_{\hat{\phi}}$, i.e., $\sup_{t\in\mathbb{Z}_{N-1},k\in\mathbb{Z}_+}\left|\hat{\boldsymbol{\phi}}_{k,k-1}(t)\right| \leq \beta_{\hat{\phi}}$.

In summary, the new set-point learning control algorithm named PI-ADDSPLC is given as Algorithm 2.

In Algorithm 2, ε_1 and ε_2 are two small positive thresholds specified in practice. Note that the resetting algorithms for $\hat{\boldsymbol{\phi}}_{k,k-1}(t)$ and $\theta_k(t)$ are used to improve the parameter estimation performance, in particular for estimating time-varying parameters, as studied in the recent papers [27] and [29].

Remark 4 *The newly developed PI-ADDSLC scheme can be extended to the design of a PID-type ADDSLC scheme, by rewriting a PID controller of the form* $u_k(t) = \tau_P e^s_k(t) + \tau_I\sum_{i=0}^{t} e^s_k(i) + \tau_D[e^s_k(t) - e^s_k(t-1)]$ *as* $u_k(t) = (\tau_P+\tau_I+\tau_D)e^s_k(t) + (\tau_I-\tau_D)e^s_k(t-1) + \sum_{i=0}^{t-2} e^s_k(i)$*. Then the above design can be straightforwardly applied, and hence the details are omitted.*

5 Robust convergence and boundedness analysis of the PI-ADDSPLC scheme

Nonrepetitive initial conditions and/or disturbances hinder or even jeopardize the convergence of a data-driven set-point learning control scheme for a batch run. Moreover, it remains an open problem in the literature to analyze the boundedness of set-point command and adaptive learning gain, without the process modeling or any prior knowledge of the batch run. To ensure robust

Algorithm 2 (PI-ADDSPLC)

Input: Initial learning gain $\theta_0(t)$, gradient vector $\hat{\boldsymbol{\phi}}_0(t)$, control input $u_0(t)$, process output $y_0(t)$, desired reference trajectory y_{d}, PI controller parameters τ_{P} and τ_{I}, tuning parameters η, μ, ρ and α, weighting factor λ, user specified thresholds ε_1 and ε_2, batch length N and maximum batch number $k_{\max}$.

Output: Process output and ATE index.

```
for k = 1, 2, ..., k_max do
  for t = 1, 2, ..., N do
    Update the gradient vector φ̂_{k,k−1}(t) by (20). If ||φ̂_{k,k−1}(t)|| ≤ ε_1 or sign(φ̂_{k,k−1}(t)) ≠
    sign(φ̂_{k,k−1}(0)), φ̂_{k,k−1}(t) is reset as its initial value φ̂_0(t);
    Update the adaptive learning gain θ_k(t) by (19) with φ^t_{k,k−1}(i), i = 0, 1, ..., t replaced by
    its estimation φ̂^t_{k,k−1}(i) and χ_k(t) set as zero. If |θ_k(t)| ≤ ε_2 or sign(θ_k(t)) ≠ sign(θ_0(t)),
    θ_k(t) is reset as its initial value θ_0(t);
    Update the set-point command y^s_k(t) by (15) and compute the set-point tracking error
    e^s_k(t) by (14);
    Apply the PI controller in (13) to a batch process in (1);
  end for
  Compute the output tracking error e_k(t) and ATE index defined by ATE(k) =
  ∑_{t=1}^{N} |e_k(t)|/N;
end for
```

convergence of output tracking error by the new PI-ADDSPLC scheme, sufficient conditions are therefore established in the following theorem, which also guarantee the boundedness of set-point command and adaptive learning gain along the batch direction.

Theorem 1 *Consider a batch process described by* (1) *controlled by application of Algorithm 2 using* (13) *and* (15) *under Assumptions 1, 2 and 4. Then the bounded tracking objective in* (2)*, the bounded set-point command in* (3)*, and the bounded adaptive learning gain in* (15) *hold, if the tuning parameter α is properly taken such that*

$$|\tau\alpha| < \frac{2}{\beta_\phi \iota}, \tag{21}$$

$$sign(\tau\alpha) = sign(\theta_0(t)), \tag{22}$$

where $\iota < \infty$ is a uniform and attainable upper bound of the learning gain $\theta_k(t)$ for any $t \in \mathbb{Z}_T$ and $k \in \mathbb{Z}_+$.

Rewriting (18) gives

$$e_k(t+1) = \mathcal{A}^e_k(t) e_{k-1}(t+1) + r_{1,k}(t), \tag{23}$$

where $\mathcal{A}^e_k(t) \triangleq 1 - \tau\alpha\phi^t_{k,k-1}(t)\theta_k(t)$ and

$$\begin{aligned} r_{1,k}(t) \triangleq & -\sum_{j=0}^{t-1} \phi^t_{k,k-1}(j)\Delta u_k(j) + \tau\phi^t_{k,k-1}(t)\Delta y_k(t) \\ & - \tau_{\mathrm{I}}\phi^t_{k,k-1}(t)\sum_{j=0}^{t-1}(\Delta y^s_k(j) - \Delta y_k(j)) - \chi_k(t). \end{aligned}$$

Setting $i = k-1$ and $j = 0$ in (5) leads to

$$\begin{aligned} y_{k-1}(t+1) = {} & y_0(t+1) + \phi^t_{k-1,0}(t)u_{k-1}(t) \\ & + \sum_{j=0}^{t-1} \phi^t_{k-1,0}(j)u_{k-1}(j) - \sum_{j=0}^{t} \phi^t_{k-1,0}(j)u_0(j) \\ & + \sum_{j=0}^{t-1} \upsilon^t_{k-1,0}(j)[\omega_{k-1}(j) - \omega_0(j)] \\ & + [\omega_{k-1}(t) - \omega_0(t)] + \vartheta^t_{k-1,0}(\delta_{k-1} - \delta_0). \end{aligned} \tag{24}$$

Substituting (24) into the set-point updating law in (15) gives

$$y^s_k(t) = \mathcal{A}^s_k(t) y^s_{k-1}(t) + r_{2,k}(t), \tag{25}$$

where $\mathcal{A}^s_k(t) \triangleq 1 - \alpha\tau\theta_k(t)\phi^t_{k-1,0}(t)$ and

$$\begin{aligned} r_{2,k}(t) \triangleq \alpha\theta_k(t) \Bigg\{ & e_0(t+1) + \tau\phi^t_{k-1,0}(t)y_{k-1}(t) \\ & - \tau_{\mathrm{I}}\phi^t_{k-1,0}(t) \sum_{i=0}^{t-1} [y^s_{k-1}(i) - y_{k-1}(i)] \\ & - \sum_{j=0}^{t-1} \phi^t_{k-1,0}(j)u_{k-1}(j) + \sum_{j=0}^{t} \phi^t_{k-1,0}(j)u_0(j) \\ & - [\omega_{k-1}(t) - \omega_0(t)] - \vartheta^t_{k-1,0}(\delta_{k-1} - \delta_0) \\ & - \sum_{j=0}^{t-1} \upsilon^t_{k-1,0}(j)[\omega_{k-1}(j) - \omega_0(j)] \Bigg\}. \end{aligned}$$

Meanwhile, by substituting $\phi^t_{k,k-1}(t)$ with its estimation $\hat{\phi}^t_{k,k-1}(t)$ and letting $\chi_k(t) = 0$, it follows from (19) that

$$\begin{aligned} \theta_k(t) = {} & \left[1 - \frac{\rho\hat{\nu}^2_k(t)}{\lambda + \hat{\nu}^2_k(t)} \right] \theta_{k-1}(t) \\ & + \frac{\rho\hat{\nu}_k(t)}{\lambda + \hat{\nu}^2_k(t)} e_{k-1}(t+1) + r_{3,k}(t), \end{aligned} \tag{26}$$

where

$$\begin{aligned} r_{3,k}(t) \triangleq \frac{\rho\hat{\nu}_k(t)}{\lambda + \hat{\nu}^2_k(t)} \Bigg\{ & - \tau_{\mathrm{I}}\hat{\phi}^t_{k,k-1}(t) \sum_{i=0}^{t-1} [\Delta y^s_k(i) - \Delta y_k(i)] \\ & - \sum_{i=0}^{t-1} \hat{\phi}^t_{k,k-1}(i)\Delta u_k(i) + \tau\hat{\phi}^t_{k,k-1}(t)\Delta y_k(t) \Bigg\}. \end{aligned}$$

Next, the mathematical induction is adopted to prove the bounded convergence of the output tracking error $e_k(t)$ along the batch direction, together with the boundedness of the adaptive learning gain $\theta_k(t)$ and the real-time updated set-point command $y^s_k(t)$ for any $t \in \mathbb{Z}_{N-1}$ and $k \in \mathbb{Z}_+$.

Step (I). When $t=0$, there is

$$\begin{aligned}|r_{3,k}(0)| &= \left|\frac{\rho\hat{\nu}_k(0)}{\lambda+\hat{\nu}_k^2(0)}\tau\hat{\phi}_{k,k-1}^t(0)\Delta y_k(0)\right| \\ &\le \frac{\rho}{2\sqrt{\lambda}}|\tau|\beta_{\hat{\phi}}\beta_{\Delta\delta}<\infty.\end{aligned}$$

Moreover, the boundedness of the second term in the right-hand side of (26) holds for any t and k since

$$\begin{aligned}\left|\frac{\rho\hat{\nu}_k(t)}{\lambda+\hat{\nu}_k^2(t)}e_{k-1}(t+1)\right| &\le \left|\frac{\rho\tau\alpha\hat{\phi}_{k,k-1}^t(t)e_{k-1}^2(t+1)}{\lambda+[\tau\alpha\hat{\phi}_{k,k-1}^t(t)e_{k-1}(t+1)]^2}\right| \\ &\le \left|\frac{\rho\tau\alpha\hat{\phi}_{k,k-1}^t(t)e_{k-1}^2(t+1)}{[\tau\alpha\hat{\phi}_{k,k-1}^t(t)e_{k-1}(t+1)]^2}\right| \\ &= \left|\frac{\rho}{\tau\alpha\hat{\phi}_{k,k-1}^t(t+1)}\right| \le \frac{\rho}{\tau|\alpha|\varepsilon_1}.\end{aligned} \tag{27}$$

Owing to $\rho\in(0,2)$, it follows that

$$\left|1-\frac{\rho\hat{\nu}_k^2(t)}{\lambda+\hat{\nu}_k^2(t)}\right|<1,\ \forall t\in\mathbb{Z}_{N-1},\ k\in\mathbb{Z}_+. \tag{28}$$

Also, it follows from (26) that $\theta_k(0)$ is bounded for any $k\in\mathbb{Z}_+$, satisfying $\sup_{k\in\mathbb{Z}_+}|\theta_k(0)|\le\beta_\theta(0,\bullet)<\infty$, where $\beta_\theta(0)>0$ is a finite constant.

It is easily verified that

$$\begin{aligned}|r_{1,k}(0)| &= \left|\tau\phi_{k,k-1}^t(0)\Delta y_k(0)-\chi_k(0)\right| \\ &\le |\tau|\beta_\phi\beta_{\Delta\delta}+\beta_\chi(\bullet)<\infty.\end{aligned}$$

By selecting $\iota=\beta_\theta(0,\bullet)$, it follows from the conditions in (21) and (22) that $|\mathcal{A}_k^e(0)|<1$ for any $k\in\mathbb{Z}_+$, which implies that the bounded convergence of $e_k(1)$ along the batch direction is guaranteed and satisfies

$$\sup_{k\in\mathbb{Z}_+}|e_k(1)|\le\beta_e(1,\bullet),$$

$$\limsup_{k\to\infty}|e_k(1)|\le\beta_{e_{\sup}}(1,\bullet),$$

where $\beta_e(1,\bullet)$ and $\beta_{e_{\sup}}(1,\bullet)$ with $\beta_e(1,\bullet)>\beta_{e_{\sup}}(1,\bullet)>0$ are two finite constants related to the upper bound of $r_{1,k}(0)$ and hence are dependent on $\beta_{\Delta\omega}$ and $\beta_{\Delta\delta}$.

Consequently, there are

$$\begin{aligned}|\Delta y_k^s(0)| &= |\alpha\theta_k(0)e_{k-1}(1)| \\ &\le |\alpha|\beta_\theta(0,\bullet)\beta_e(1,\bullet)<\infty\end{aligned}$$

and

$$\begin{aligned}|\Delta u_k(0)| &= |\tau(\Delta y_k^s(0)-\Delta y_k(0))| \\ &\le |\tau|\big[|\alpha|\beta_\theta(0,\bullet)\beta_e(1,\bullet)+\beta_{\Delta\delta}\big]<\infty.\end{aligned}$$

Based on the boundedness of $\theta_k(0)$ and $e_k(1)$, it follows by using Assumptions 1, 4 and Lemma 1 that

$$\begin{aligned}|r_{2,k}(0)| &= \Big|\alpha\theta_k(0)\Big\{e_0(1) + \tau\phi^t_{k-1,0}(0)y_{k-1}(0)\\ &\quad + \phi^t_{k-1,0}(0)u_0(0) - [\omega_{k-1}(t) - \omega_0(0)]\\ &\quad - \vartheta^t_{k-1,0}(\delta_{k-1} - \delta_0)\Big\}\Big|\\ &\le |\alpha|\beta_\theta(0,\bullet)\Big\{\beta_e(1,\bullet) + |\tau|\beta_\phi(|y_0| + \beta_\delta)\\ &\quad + 2(\beta_\omega + \beta_\phi\beta_\delta)\Big\} < \infty, \quad \forall k \in \mathbb{Z}_+.\end{aligned}$$

Using the conditions in (21) and (22) again with $\iota = \beta_\theta(0,\bullet)$, it follows that $|\mathcal{A}^s_k(0)| < 1$ for any $k \in \mathbb{Z}_+$, and the boundedness of $y^s_k(0)$ follows immediately, i.e., $\sup_{k\in\mathbb{Z}_+} |y^s_k(0)| \le \beta_s(0,\bullet) < \infty$, where $\beta_s(0,\bullet) > 0$ is a finite constant.

Step (II). Suppose that for any $t \in \mathbb{Z}_{T-1}$ with $T \in \mathbb{Z}_{N-1}$, the following conditions hold

$$\sup_{k\in\mathbb{Z}_+} |\theta_k(t)| \le \beta_\theta(t,\bullet) < \infty, \tag{29}$$

$$\sup_{k\in\mathbb{Z}_+} |e_k(t+1)| \le \beta_e(t+1,\bullet) < \infty, \tag{30}$$

$$\limsup_{k\in\mathbb{Z}_+} |e_k(t+1)| \le \beta_{e_{\sup}}(t+1,\bullet) < \infty, \tag{31}$$

$$\sup_{k\in\mathbb{Z}_+} |y^s_k(t)| \le \beta_s(t,\bullet) < \infty, \tag{32}$$

where $\beta_\theta(t,\bullet)$, $\beta_s(t,\bullet)$, $\beta_e(t+1,\bullet)$ and $\beta_{e_{\sup}}(t+1,\bullet)$ for any $t \in \mathbb{Z}_{T-1}$ are finite constants satisfying $\beta_e(t+1,\bullet) > \beta_{e_{\sup}}(t+1,\bullet) > 0$ and dependent on $\beta_{\Delta\omega}$ and $\beta_{\Delta\delta}$.

When it comes to the time $t = T$, it follows that

$$\begin{aligned}|r_{3,k}(T)| &\le \frac{\rho|\hat{\nu}_k(T)|}{2\sqrt{\lambda}|\hat{\nu}_k(T)|}\Bigg|\tau_{\mathrm{I}}\hat{\phi}^t_{k,k-1}(T)\sum_{i=0}^{T-1}[\Delta y^s_k(t) - \Delta y_k(i)]\\ &\quad + \sum_{i=0}^{T-1}\hat{\phi}^t_{k,k-1}(i)\Delta u_k(i) - \tau\hat{\phi}^t_{k,k-1}(T)\Delta y_k(T)\Bigg|\\ &\le \frac{\rho}{2\sqrt{\lambda}}\Bigg\{|\tau_{\mathrm{I}}|\beta_{\hat{\phi}}T\max_{i\in\mathbb{Z}_{T-1}}\big[|\Delta y^s_k(i)| + |\Delta y_k(i)|\big]\\ &\quad + T\beta_{\hat{\phi}}\max_{i\in\mathbb{Z}_{T-1}}|\Delta u_k(i)| + |\tau|\beta_{\hat{\phi}}|\Delta y_k(T)|\Bigg\}.\end{aligned}$$

Additionally, it follows from (29)-(32) that

$$\begin{aligned}
\max_{i\in\mathbb{Z}_{T-1}} |\Delta y_k^s(i)| &\le 2\beta_{s,T-1}^{\max}(\bullet),\\
\max_{i\in\mathbb{Z}_{T-1}} |\Delta y_k(i)| &\le 2\beta_{e,T-1}^{\max}(\bullet),\\
\max_{i\in\mathbb{Z}_{T-1}} |\Delta u_k(i)| &\le 2|\tau| \max_{i\in\mathbb{Z}_{T-1}} \{\beta_s(i,\bullet)+\beta_e(i,\bullet)\}\\
&\quad + \max_{i\in\mathbb{Z}_{T-1}} 2|\tau_{\mathrm{I}}| i \max_{j\in\mathbb{Z}_{i-1}} \{\beta_s(j,\bullet)+\beta_e(j,\bullet)\}\\
&\le 2\big[|\tau|+|\tau_{\mathrm{I}}|(T-1)\big]\\
&\quad \times \left(\beta_{s,T-1}^{\max}(\bullet)+\beta_{e,T-1}^{\max}(\bullet)\right),
\end{aligned}$$

where

$$\beta_{s,T-1}^{\max}(\bullet) \triangleq \max_{i\in\mathbb{Z}_{T-1}} \beta_s(i,\bullet),\ \ \beta_{e,T-1}^{\max}(\bullet) \triangleq \max_{i\in\mathbb{Z}_{T-1}} \beta_e(i,\bullet).$$

Therefore

$$\begin{aligned}
|r_{3,k}(T)| \le \frac{\rho}{2\sqrt{\lambda}} \Bigg[& 2|\tau_{\mathrm{I}}|\beta_{\hat{\phi}} T(\beta_{s,T-1}^{\max}(\bullet)+\beta_{e,T-1}^{\max}(\bullet))\\
&+ 2|\tau|\beta_{\hat{\phi}}\beta_e(T,\bullet) + 2T\beta_{\hat{\phi}}\big[|\tau|+|\tau_{\mathrm{I}}|(T-1)\big]\\
&\times (\beta_{s,T-1}^{\max}(\bullet)+\beta_{e,T-1}^{\max}(\bullet))\Bigg] < \infty,
\end{aligned} \tag{33}$$

and hence $r_{3,k}(T)$ is bounded for any $k\in\mathbb{Z}_+$, and its upper bound is related to $\beta_{\Delta\omega}$ and $\beta_{\Delta\delta}$.

Based on the inequalities (27) and (28), it follows from (26) and (33) that $\sup_{k\in\mathbb{Z}_+} |\theta_k(T)| \le \beta_\theta(T,\bullet) < \infty$, where $\beta_\theta(T,\bullet)$ is a finite constant. Consequently, it is derived that

$$\begin{aligned}
|r_{1,k}(T)| &\le \beta_\phi \sum_{j=0}^{T-1} |\Delta u_k(j)| + |\tau|\beta_\phi|\Delta y_k(T)| + |\chi_k(T)|\\
&\quad + |\tau_{\mathrm{I}}|\beta_\phi \sum_{j=0}^{T-1} \big[|\Delta y_k^s(j)+|\Delta y_k(j)|\big]\\
&\le 2\beta_\phi T(|\tau|+|\tau_{\mathrm{I}}|(T-1))\\
&\quad \times (\beta_{s,T-1}^{\max}(\bullet)+\beta_{e,T-1}^{\max}(\bullet)) + 2|\tau|\beta_\phi\beta_e(T,\bullet)\\
&\quad + 2|\tau_{\mathrm{I}}|\beta_\phi T\left(\beta_{s,T-1}^{\max}(\bullet)+\beta_{e,T-1}^{\max}(\bullet)\right) + \beta_\chi < \infty.
\end{aligned}$$

Similarly, using the conditions in (21) and (22) with $\iota = \beta_\theta(T,\bullet)$ gives $|\mathcal{A}_k^e(T)| < 1$ for any $k\in\mathbb{Z}_+$, which guarantees the bounded convergence of $e_k(T+1)$ for any $k\in\mathbb{Z}_+$, i.e.

$$\begin{aligned}
\sup_{k\in\mathbb{Z}_+} e_k(T+1) &\le \beta_e(T+1,\bullet),\\
\limsup_{k\to\infty} |e_k(T+1)| &\le \beta_{e_{\sup}}(T+1,\bullet) < \beta_e(T+1,\bullet),
\end{aligned}$$

where $\beta_e(T+1,\bullet)$ and $\beta_{e_{\sup}}(T+1,\bullet)$ are two finite positive constants dependent on $\beta_{\Delta\omega}$ and $\beta_{\Delta\delta}$.

Hence, there are

$$\begin{aligned}
|\Delta y_k^s(T)| &= |\alpha\theta_k(T)e_{k-1}(T+1)|\\
&\le |\alpha|\beta_\theta(T,\bullet)\beta_e(T+1,\bullet) < \infty
\end{aligned}$$

and

$$
\begin{aligned}
|\Delta u_k(T)| &= \left| \tau \Delta e_k^s(T) + \tau_{\mathrm{I}} \sum_{i=0}^{T-1} \Delta e_k^s(i) \right| \\
&\leq |\tau| \left[|\alpha| \beta_\theta(T, \bullet) \beta_e(T+1, \bullet) + 2\beta_e(T, \bullet) \right] \\
&\quad + 2|\tau_{\mathrm{I}}| T \left(\beta_{s,T-1}^{\max}(\bullet) + \beta_{e,T-1}^{\max}(\bullet) \right) < \infty.
\end{aligned}
$$

Using the boundedness of $y_{k-1}(t)$ for any $t \in \mathbb{Z}_T$ and $u_{k-1}(t)$ for any $t \in \mathbb{Z}_{T-1}$, respectively, i.e.

$$
\begin{aligned}
|y_{k-1}(t)| &\leq |y_{\mathrm{d}}(t)| + |e_{k-1}(t)| \leq \beta_{\mathrm{d}} + \beta_{e,T}^{\max}(\bullet), \\
|u_{k-1}(t)| &\leq |\tau| (|y_{k-1}^s(t)| + |y_{k-1}(t)|) \\
&\quad + |\tau_{\mathrm{I}}| \sum_{i=0}^{t-1} [|y_{k-1}^s(i)| + |y_{k-1}(i)|] \\
&\leq (|\tau| + (T-1)|\tau_{\mathrm{I}}|)(\beta_{s,T-1}^{\max}(\bullet) + \beta_{\mathrm{d}} + \beta_{e,T}^{\max}(\bullet)),
\end{aligned}
$$

where $\beta_{e,T}^{\max}(\bullet) \triangleq \max_{i \in \mathbb{Z}_T} \beta_e(i, \bullet)$, it follows that

$$
\begin{aligned}
|r_{2,k}(T)| \leq |\alpha| \beta_\theta(T, \bullet) \Big[& \beta_e(T+1, \bullet) + |\tau| \beta_\phi (\beta_{\mathrm{d}} + \beta_{e,T}^{\max}(\bullet)) \\
& + \beta_\phi |\tau_{\mathrm{I}}| T (\beta_{s,T-1}^{\max}(\bullet) + \beta_{\mathrm{d}} + \beta_{e,T}^{\max}(\bullet)) \\
& + \beta_\phi T (|\tau| + (T-1)|\tau_{\mathrm{I}}|) \\
& \times (\beta_{s,T-1}^{\max}(\bullet) + \beta_{\mathrm{d}} + \beta_{e,T}^{\max}(\bullet)) \\
& + 2(\beta_\omega + T\beta_\phi\beta_\omega + \beta_\phi\beta_\delta) \Big] < \infty,
\end{aligned}
$$

which, together with the conditions in (21) and (22) with $\iota = \beta_\theta(T, \bullet)$, guarantees $|\mathcal{A}_k^s(T)| < 1$ for any $k \in \mathbb{Z}_+$. Therefore, $y_k^s(T)$ is bounded and satisfies $\sup_{k \in \mathbb{Z}_+} |y_k^s(T)| \leq \beta_s(T, \bullet)$, where $\beta_s(T, \bullet) > 0$ is a finite constant.

Hence, according to the mathematical induction, the conclusion of this theorem is true. Since $t \in \mathbb{Z}_T$ is finite, a uniform threshold of $\iota = \max_{i \in \mathbb{Z}_T} \beta_\theta(i, \bullet)$ can be taken. This completes the proof.

Remark 5 *The parameter α plays a crucial role in the conditions of* (21) *and* (22) *in Theorem 1, which is the only tuning parameter for the set-point learning control since both $\boldsymbol{\phi}_{k,k-1}^t(t)$ and $\theta_k(t)$ are iteratively estimated. Moreover, the bounded control input required by the developed DDILC methods (e.g., [25, 26, 27, 28]) is obviously guaranteed by the new PI-ADDSPLC scheme, based on the above designed PI controller in* (13) *and the boundedness of output tracking error and the set-point command as concluded by Theorem 1.*

If both nonrepetitive disturbances and initial condition shifts converge along the batch direction, i.e.

$$
\begin{aligned}
\lim_{k \to \infty} [\omega_k(t) - \omega_{k-1}(t)] = 0, \ \forall t \in \mathbb{Z}_{N-1}, \\
\lim_{k \to \infty} (\delta_k - \delta_{k-1}) = 0,
\end{aligned}
\tag{34}
$$

perfect output tracking could be realized despite the presence of nonrepetitive disturbances and initial conditions, which is clarified by the following corollary.

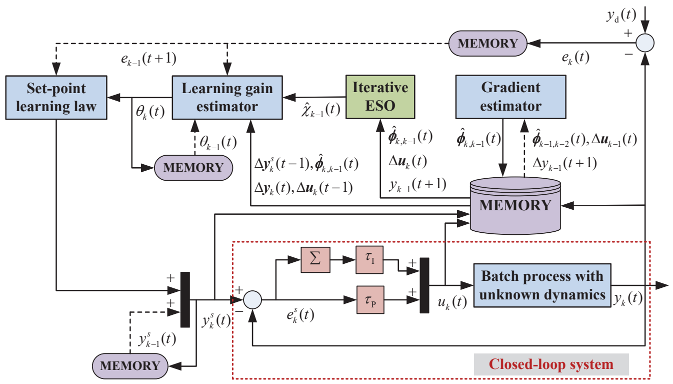

Figure 2: Schematic of the new IESO based PI-ADDSPLC scheme

Corollary 1 *Consider a batch process described by* (1) *controlled by application of Algorithm 2 using* (13) *and* (15) *under Assumptions 1, 2 and the condition in* (34)*. Then, asymptotic convergence of the output tracking error together with the boundedness of the real-time updated set-point command in* (3) *and the adaptive learning gain in* (15) *is guaranteed, if the tuning parameter* α *is taken such that the conditions in* (21) *and* (22) *are satisfied.*

Using (34), it follows from (7) that

$$\lim_{k\to\infty} \chi_k(t) = 0. \tag{35}$$

The rest of the proof follows similar steps to those for Theorem 1, and therefore is omitted.

6 IESO based PI-ADDSPLC scheme

Although the PI-ADDSPLC scheme developed in Section 4 has the capability to maintain robust convergence and boundedness of the resulting ILC system, the adverse effects caused by nonrepetitive uncertainties cannot be actively suppressed. Motivated by the time-domain ESO design in ADRC [34], another IESO based PI-ADDSPLC scheme shown in Fig. 2 is further developed in this section by inserting an IESO into the PI-ADDSPLC scheme to proactively estimate and counteract $\chi_k(t)$ that was not considered in the PI-ADDSPLC scheme, for better disturbance rejection but at the cost of implemental complexity. Note that the adopted IESO can estimate the uncertainty term $\chi_k(t)$ along the batch direction, by only using the available process data as shown in Fig. 2 rather than any model information as required in other recent research [38].

For analysis, the following assumption is made without loss of generality.

Assumption 5 *The variation of the set-point command* $y_k^s(t)$ *along the batch direction is bounded, i.e.,* $|\Delta y_k^s(t)| \le \delta_{sp}$ *for any* $t \in \mathbb{Z}_{N-1}$*,* $k \in \mathbb{Z}_+$*, where* $\delta_{sp} > 0$ *is a finite constant.*

By augmenting the so-called *total disturbance* $\chi_k(t)$ as an extended state in (7), it follows that

$$\begin{cases} \mathcal{Y}_k(t+1) = A\mathcal{Y}_{k-1}(t+1) + \mathcal{I}\boldsymbol{\phi}^{\top}_{k,k-1}(t)\Delta\boldsymbol{u}_k(t) + \Omega_k(t), \\ y_{k-1}(t+1) = C\mathcal{Y}_{k-1}(t+1), \end{cases} \tag{36}$$

where $C = [1\ \ 0]$, $\mathcal{Y}_k(t+1) = [y_k(t+1)\ \ \chi_k(t)]^\top$ and

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix},\ \mathcal{I} = \begin{bmatrix} 1 \\ 0 \end{bmatrix},\ \Omega_k(t) = \begin{bmatrix} \Delta\chi_k(t) \\ \Delta\chi_k(t) \end{bmatrix}.$$

Note that $\Omega_k(t)$ is bounded for any $t \in \mathbb{Z}_{N-1}$ and $k \in \mathbb{Z}_+$ due to the boundedness of $\chi_k(t)$, and is assumed to satisfy $\sup_{t\in\mathbb{Z}_{N-1},k\in\mathbb{Z}_+} |\Omega_k(t)| \le \beta_\Omega(\bullet)$, where $\beta_\Omega(\bullet)$ is a finite constant dependent on $\beta_{\Delta\omega}$ and $\beta_{\Delta\delta}$.

An IESO for the augmented system in (36) is designed as

$$\begin{aligned} \hat{\mathcal{Y}}_k(t+1) = A\hat{\mathcal{Y}}_{k-1}(t+1) + \mathcal{I}\hat{\boldsymbol{\phi}}_{k,k-1}^\top(t)\Delta\boldsymbol{u}_k(t) \\ + L\left(y_{k-1}(t+1) - C\hat{\mathcal{Y}}_{k-1}(t+1)\right), \end{aligned} \tag{37}$$

where $L = [l_1\ l_2]^\top$ is the observer gain vector, $\hat{\mathcal{Y}}_{k-1}(t+1) = [\hat{y}_{k-1}(t+1)\ \ \hat{\chi}_{k-1}(t)]^\top$ is the estimation of $\mathcal{Y}_{k-1}(t+1)$, and $\hat{\boldsymbol{\phi}}_{k,k-1}(t)$ is the estimation of $\boldsymbol{\phi}_{k,k-1}(t)$ obtained from the parameter estimation in (20).

By substituting $\phi_{k,k-1}^t(i), i = 0, 1, \ldots, t$ and $\chi_k(t)$ with $\hat{\phi}_{k,k-1}^t(i)$ and $\hat{\chi}_k(t)$ in (19), a new updating law for the adaptive learning gain is established below

$$\begin{aligned} \theta_k(t) = \theta_{k-1}(t) + \frac{\rho\hat{\nu}_k(t)}{\lambda + \hat{\nu}_k^2(t)} \times \\ \Bigg\{ e_{k-1}(t+1) - \theta_{k-1}(t)\hat{\nu}_k(t) - \hat{\chi}_k(t) \\ - \sum_{j=0}^{t-1} \hat{\phi}_{k,k-1}^t(j)\Delta u_k(j) + \tau\hat{\phi}_{k,k-1}^t(t)\Delta y_k(t) \\ - \tau_{\mathrm{I}}\hat{\phi}_{k,k-1}^t(t)\sum_{i=0}^{t-1}(\Delta y_k^s(i) - \Delta y_k(i)) \Bigg\}. \end{aligned} \tag{38}$$

Correspondingly, the set-point learning law is established as

$$y_k^s(t) = y_{k-1}^s(t) + \begin{cases} \zeta_k(t), & \text{if } |\zeta_k(t)| \le \delta_{\mathrm{sp}}, \\ \delta_{\mathrm{sp}}\,\mathrm{sign}(\zeta_k(t)), & \text{otherwise}, \end{cases} \tag{39}$$

where $\zeta_k(t) \triangleq \alpha\theta_k(t)e_{k-1}(t+1)$ and $\mathrm{sign}(\cdot)$ is the sign function.

The new IESO based PI-ADDSPLC scheme is summarized in Algorithm 3.

The following theorem establishes sufficient conditions to ensure robust convergence of the output tracking error when Algorithm 3 is applied in the presence of nonrepetitive uncertainties.

Theorem 2 *Consider a batch process described by* (1) *controlled by application of Algorithm 3 using* (13) *and* (15) *under Assumptions 1, 2, 4 and 5. Then the bounded tracking objective in* (2)*, the bounded set-point command in* (3)*, the bounded adaptive learning gain in* (15) *and the bounded convergence of IESO in* (37) *hold, if the tuning parameter* α *is taken such that the conditions in* (21) *and* (22) *are satisfied, and the IESO gains* l_1 *and* l_2 *are selected to satisfy*

$$\max\left\{ \frac{|2-l_1+\sqrt{l_1^2-4l_2}|}{2}, \frac{|2-l_1-\sqrt{l_1^2-4l_2}|}{2} \right\} < 1. \tag{40}$$

Algorithm 3 (IESO based PI-ADDSPLC)

Input: Initial learning gain $\theta_0(t)$, gradient vector $\hat{\boldsymbol{\phi}}_0(t)$, control input $u_0(t)$, process output $y_0(t)$, IESO state $\hat{\mathcal{Y}}_0(t)$, desired reference trajectory y_{d}, PI controller parameters τ_{P} and τ_{I}, observer gains l_1 and l_2, tuning parameters η, μ, ρ and α, weighting factor λ, user specified thresholds ε_1, ε_2 and δ_{sp}, batch length N and maximum batch number $k_{\max}$.

Output: Process output and ATE index.

1: **for** $k = 1, 2, \ldots, k_{\max}$ **do**
2: **for** $t = 1, 2, \ldots, N$ **do**
3: Update the gradient vector $\hat{\boldsymbol{\phi}}_{k,k-1}(t)$ by (20). If $\|\hat{\boldsymbol{\phi}}_{k,k-1}(t)\| \le \varepsilon_1$ or $\mathrm{sign}(\hat{\boldsymbol{\phi}}_{k,k-1}(t)) \ne \mathrm{sign}(\hat{\boldsymbol{\phi}}_{k,k-1}(0))$, $\hat{\boldsymbol{\phi}}_{k,k-1}(t)$ is reset as its initial value $\hat{\boldsymbol{\phi}}_0(t)$;
4: Update the IESO state $\hat{\mathcal{Y}}_k(t+1)$ by (37);
5: Update the adaptive learning gain $\theta_k(t)$ by (38). If $|\theta_k(t)| \le \varepsilon_2$ or $\mathrm{sign}(\theta_k(t)) \ne \mathrm{sign}(\theta_0(t))$, $\theta_k(t)$ is reset as its initial value $\theta_0(t)$;
6: Update the set-point command $y_k^s(t)$ by (39) and compute the set-point tracking error $e_k^s(t)$ by (14);
7: Apply the PI controller in (13) to a batch process in (1);
8: **end for**
9: Compute the output tracking error $e_k(t)$ and ATE index;
10: **end for**

The boundedness of $\chi_k(t)$ for any $t \in \mathbb{Z}_{N-1}$ and $k \in \mathbb{Z}_+$ ensures that $\hat{\boldsymbol{\phi}}_{k,k-1}(t)$ is bounded, for which the proof is similar to that of Theorem 6 in [39] and thus is omitted.

On setting $\tilde{\mathcal{Y}}_k(t) = \mathcal{Y}_k(t) - \hat{\mathcal{Y}}_k(t)$. The dynamics of the IESO estimation error is obtained as

$$\begin{aligned}\tilde{\mathcal{Y}}_k(t+1) = (A - LC)&\tilde{\mathcal{Y}}_{k-1}(t+1)\\ &+ \mathcal{I}\tilde{\boldsymbol{\phi}}_{k,k-1}^{\top}(t)\Delta\boldsymbol{u}_k(t) + \Omega_k(t),\end{aligned} \tag{41}$$

where $\tilde{\boldsymbol{\phi}}_{k,k-1}(t) \triangleq \boldsymbol{\phi}_{k,k-1}(t) - \hat{\boldsymbol{\phi}}_{k,k-1}(t)$. Note that the boundedness of $\boldsymbol{\phi}_{k,k-1}(t)$ and $\hat{\boldsymbol{\phi}}_{k,k-1}(t)$ ensures that $\tilde{\boldsymbol{\phi}}_{k,k-1}(t)$ is bounded, and assumed to satisfy $\|\tilde{\boldsymbol{\phi}}_{k,k-1}(t)\| \le \beta_{\tilde{\phi}} < \infty$ for any $t \in \mathbb{Z}_{N-1}$ and $k \in \mathbb{Z}_+$, where $\beta_{\tilde{\phi}}$ is a finite constant.

Applying the PI control law in (13) to (41) gives

$$\tilde{\mathcal{Y}}_k(t+1) = (A - LC)\tilde{\mathcal{Y}}_{k-1}(t+1) + r_{4,k}(t), \tag{42}$$

where

$$\begin{aligned}r_{4,k}(t) = &\,\mathcal{I}\tau\tilde{\phi}_{k,k-1}^t(t)(\Delta y_k^s(t) - \Delta y_k(t))\\ &+ \mathcal{I}\tau_{\mathrm{I}}\tilde{\phi}_{k,k-1}^t(t)\sum_{i=0}^{t-1}[\Delta y_k^s(i) - \Delta y_k(t)]\\ &+ \mathcal{I}\sum_{j=0}^{t-1}\tilde{\phi}_{k,k-1}^t(j)\Delta u_k(j) + \Omega_k(t).\end{aligned}$$

Let $h_k(t) = \Delta y_k^s(t)/\zeta_k(t)$. Then it follows that $h_k(t) = 1$ if $|\zeta_k(t)| \le \delta_{\mathrm{sp}}$, and $h_k(t) \in (0,1)$ if $|\zeta_k(t)| > \delta_{\mathrm{sp}}$. Hence, the dynamics of the output tracking error along the batch direction is given by

$$e_k(t+1) = \overline{\mathcal{A}}_k^e(t)e_{k-1}(t+1) + r_{1,k}(t), \tag{43}$$

where $\overline{\mathcal{A}}^e_k(t) \triangleq 1 - \tau\alpha h_k(t)\phi^t_{k,k-1}(t)\theta_k(t)$ and $r_{1,k}(t)$ is described as per (23).

Moreover, one has

$$\begin{aligned} y^s_k(t) &= y^s_{k-1}(t) + h_k(t)\zeta_k(t) \\ &= y^s_{k-1}(t) + h_k(t)\alpha\theta_k(t)e_{k-1}(t+1) \\ &= y^s_{k-1}(t) + h_k(t)\alpha\theta_k(t)[y_{\mathrm{d}}(t+1) - y_{k-1}(t+1)], \end{aligned}$$

which can be combined with (24) to give

$$y^s_k(t) = \overline{\mathcal{A}}^s_k(t)y^s_{k-1}(t) + h_k(t)r_{2,k}(t), \tag{44}$$

where $\overline{\mathcal{A}}^s_k(t) \triangleq 1 - \alpha\tau h_k(t)\theta_k(t)\phi^t_{k-1,0}(t)$ and $r_{2,k}(t)$ is described as per (25).

It follows from (38) that the updating of $\theta_k(t)$ is given by

$$\begin{aligned} \theta_k(t) = &\left[1 - \frac{\rho\hat{\nu}^2_k(t)}{\lambda + \hat{\nu}^2_k(t)}\right]\theta_{k-1}(t) + \frac{\rho\hat{\nu}_k(t)}{\lambda + \hat{\nu}^2_k(t)}e_{k-1}(t+1) \\ &+ r_{3,k}(t) - \frac{\rho\hat{\nu}_k(t)}{\lambda + \hat{\nu}^2_k(t)}\hat{\chi}_k(t), \end{aligned} \tag{45}$$

where $r_{3,k}(t)$ is described as per (26).

As in the proof of Theorem 1, the mathematical induction is used to prove the main conclusion of this theorem as follows.

Step (I). Let $t = 0$, and it follows that

$$r_{4,k}(0) = \mathcal{I}\tau\tilde{\phi}^t_{k,k-1}(0)[\Delta y^s_k(0) - \Delta y_k(0)] + \Omega_k(0).$$

Since $\Delta y_k(0) = y_k(0) - y_{k-1}(0) = \delta_k - \delta_{k-1}$ and $\sup_{t\in\mathbb{Z}_{N-1},k\in\mathbb{Z}_+} |\Delta y^s_k(0)| \leq \delta_{\mathrm{sp}}$, one has

$$|r_{4,k}(0)| \leq |\tau|\beta_{\tilde{\phi}}(\delta_{\mathrm{sp}} + \beta_{\Delta\delta}) + \beta_{\Omega}(\bullet) < \infty. \tag{46}$$

Therefore, the boundedness convergence of $\tilde{Y}_k(1)$ is guaranteed under the condition in (40), and hence $A - LC$ is Schur stable. This fact further implies that $\hat{\chi}_k(0)$ is bounded for any $k \in \mathbb{Z}_+$, satisfying $\sup_{k\in\mathbb{Z}_+} |\hat{\chi}_k(0)| \leq \beta_{\hat{\chi}}(0,\bullet) < \infty$, where $\beta_{\hat{\chi}}(0,\bullet)$ is a finite constant related to $\beta_{\Delta\omega}$ and $\beta_{\Delta\delta}$.

By the boundedness of $\hat{\chi}_k(0)$, it follows that

$$\begin{aligned} &\left| r_{3,k}(0) - \frac{\rho\hat{\nu}_k(0)}{\lambda + \hat{\nu}^2_k(0)}\hat{\chi}_k(0) \right| \\ = &\left| \frac{\rho\hat{\nu}_k(0)}{\lambda + \hat{\nu}^2_k(0)}\left[\tau\hat{\phi}^t_{k,k-1}(0)\Delta y_k(0) - \hat{\chi}_k(0)\right] \right| \\ \leq &\frac{\rho}{2\sqrt{\lambda}}\left(\beta_{\hat{\chi}}(0,\bullet) + |\tau|\beta_{\hat{\phi}}\beta_{\Delta\delta}\right) < \infty, \end{aligned}$$

which, together with the boundeness of $\frac{\rho\hat{\nu}_k(t)}{\lambda+\hat{\nu}^2_k(t)}e_{k-1}(t+1)$, see Theorem 1, and the condition in (28), guarantees that $\theta_k(0)$ is bounded and thus is assumed to satisfy $\sup_{k\in\mathbb{Z}_+} |\theta_k(0)| \leq \beta_\theta(0,\bullet) < \infty$, where $\beta_\theta(0,\bullet)$ is a finite constant related to $\beta_{\Delta\omega}$ and $\beta_{\Delta\delta}$. Hence

$$
\begin{aligned}
|r_{1,k}(0)| &\le |\tau\phi^t_{k,k-1}(0)\Delta y_k(0) - \chi_k(0)| \\
&\le \beta_\chi(\bullet) + |\tau|\beta_\phi\beta_{\Delta\delta} < \infty, \quad \forall k \in \mathbb{Z}_+.
\end{aligned}
$$

Using the conditions given in (21) and (22) with $\iota = \beta_\theta(0, \bullet)$, it follows that $|\overline{\mathcal{A}}^e_k(0)| < 1$ for any $k \in \mathbb{Z}_+$, which guarantees that $e_k(1)$ is bounded and convergent along the batch direction, i.e.

$$
\begin{aligned}
\sup_{k\in\mathbb{Z}_+} |e_k(1)| &\le \beta_e(1, \bullet), \\
\limsup_{k\to\infty} |e_k(1)| &\le \beta_{e_{\sup}}(1, \bullet) < \beta_e(1, \bullet),
\end{aligned}
$$

where $\beta_e(1, \bullet)$ and $\beta_{e_{\sup}}(1, \bullet)$ are two finite constants related to $\beta_{\Delta\omega}$ and $\beta_{\Delta\delta}$. As a consequence, it follows that

$$
\begin{aligned}
|h_k(0)r_{2,k}(0)| = \Big|&\alpha h_k(0)\theta_k(0)\Big\{e_0(1) + \tau\phi^t_{k-1,0}(0)y_{k-1}(0) \\
&+ \phi^t_{k-1,0}(0)u_0(0) - [\omega_{k-1}(0) - \omega_0(0)] \\
&-\vartheta^t_{k-1,0}(\delta_{k-1} - \delta_0)\Big\}\Big| \\
\le\ &|\alpha|\beta_\theta(0, \bullet)\Big[\beta_e(1, \bullet) + |\tau|\beta_\phi(|y_0| + \beta_\delta) \\
&+ 2\beta_w + 2\beta_\phi\beta_\delta\Big] < \infty.
\end{aligned}
$$

Using the conditions in (21) and (22) with $\iota = \beta_\theta(0, \bullet)$ again, it follows that $|\overline{\mathcal{A}}^s_k(0)| < 1$ for any $k \in \mathbb{Z}_+$. Therefore, $y^s_k(0)$ is guaranteed to be bounded and thus is assumed to satisfy $\sup_{k\in\mathbb{Z}_+} |y^s_k(0)| \le \beta_s(0, \bullet) < \infty$, where $\beta_s(0, \bullet)$ is a finite constant related to $\beta_{\Delta\omega}$ and $\beta_{\Delta\delta}$.

Step (II). Suppose that for any $t \in \mathbb{Z}_{T-1}$ with $T \in \mathbb{Z}_{N-1}$, the bounded convergence of $\tilde{\mathcal{Y}}_k(t)$, $\theta_k(t)$, $e_k(t + 1)$ and $y^s_k(t)$ is satisfied and

$$
\begin{aligned}
\sup_{k\in\mathbb{Z}_+} |\tilde{\mathcal{Y}}_k(t)| &\le \beta_{\tilde{\mathcal{Y}}}(t, \bullet) < \infty, \\
\sup_{k\in\mathbb{Z}_+} |\theta_k(t)| &\le \beta_\theta(t, \bullet) < \infty, \\
\sup_{k\in\mathbb{Z}_+} |e_k(t + 1)| &\le \beta_e(t + 1, \bullet) < \infty, \\
\limsup_{k\to\infty} |e_k(t + 1)| &\le \beta_{e_{\sup}}(t + 1, \bullet) < \beta_e(t + 1, \bullet), \\
\sup_{k\in\mathbb{Z}_+} |y^s_k(t)| &\le \beta_s(t, \bullet) < \infty,
\end{aligned}
$$

where $\beta_{\tilde{\mathcal{Y}}}(t, \bullet)$, $\beta_\theta(t, \bullet)$, $\beta_e(t + 1, \bullet)$, $\beta_{e_{\sup}}(t + 1, \bullet)$ and $\beta_s(t, \bullet)$ are finite constants related to $\beta_{\Delta\omega}$ and $\beta_{\Delta\delta}$.

When it comes to the time $t = T$, one has

$$
\begin{aligned}
|r_{4,k}(T)| \le\ &|\tau|\beta_{\tilde{\phi}}(\delta_{\text{sp}} + 2\beta^{\max}_{e,T}(\bullet)) + \beta_\Omega(\bullet) \\
&+ 2|\tau_{\text{I}}|\beta_{\tilde{\phi}}T(\beta^{\max}_{s,T-1}(\bullet) + \beta^{\max}_{e,T-1}(\bullet)) \\
&+2\beta_{\tilde{\phi}}T(|\tau|+|\tau_{\text{I}}|T)(\beta^{\max}_{s,T-1}(\bullet)+\beta^{\max}_{e,T-1}(\bullet)) < \infty,
\end{aligned}
$$

where $\beta^{\max}_{e,T-1}(\bullet) \triangleq \max_{t\in\mathbb{Z}_{T-1}} \beta_e(t, \bullet)$, and hence the bounded convergence of $\tilde{\mathcal{Y}}_k(T + 1)$ is guaranteed. In turn, $\hat{\chi}_k(T)$ is bounded and is assumed to satisfy $\sup_{k\in\mathbb{Z}_+} |\hat{\chi}_k(T)| \le \beta_{\hat{\chi}}(T, \bullet)$, where $\beta_{\hat{\chi}}(T,)$ is a finite constant related to $\beta_{\Delta\omega}$ and $\beta_{\Delta\delta}$.

Moreover, it follows from (45) that

$$\begin{aligned}
&\left|r_{3,k}(t)-\frac{\rho\hat{\nu}_k(t)}{\lambda+\hat{\nu}_k^2(t)}\hat{\chi}_k(t)\right|\\
&\leq\frac{\rho}{2\sqrt{\lambda}}\Big\{\beta_{\hat{\chi}}(T,\bullet)+2|\tau|\beta_{\hat{\phi}}\beta_e(T,\bullet)+2T\beta_{\hat{\phi}}\\
&\quad\times(\beta_{s,T-1}^{\max}(\bullet)+\beta_{e,T-1}^{\max}(\bullet))\left[|\tau|+(1+T)|\tau_{\mathrm{I}}|\right]\Big\}<\infty,
\end{aligned}$$

which, together with the boundedness of $\frac{\rho\hat{\nu}_k(t)}{\lambda+\hat{\nu}_k^2(t)}e_{k-1}(t+1)$, ensures the boundedness of $\theta_k(T)$, satisfying $\sup_{k\in\mathbb{Z}_+}|\theta_k(T)|\leq\beta_\theta(T,\bullet)<\infty$, where $\beta_\theta(T,\bullet)$ is a finite constant. Hence

$$\begin{aligned}
|r_{1,k}(T)|\leq&\Bigg|-\sum_{j=0}^{T-1}\phi_{k,k-1}^t(j)\Delta u_k(j)\\
&-\chi_k(T)+\tau\phi_{k,k-1}^t(T)\Delta y_k(T)\\
&-\tau_{\mathrm{I}}\phi_{k,k-1}^t(T)\sum_{i=0}^{T-1}[\Delta y_k^s(i)-\Delta y_k(i)]\Bigg|\\
\leq&\,2T\beta_\phi(|\tau|+|\tau_{\mathrm{I}}|T)(\beta_{s,T-1}^{\max}(\bullet)+\beta_{e,T-1}^{\max}(\bullet))\\
&+\beta_\chi+2|\tau|\beta_\phi\beta_e(T,\bullet)\\
&+2|\tau_{\mathrm{I}}|\beta_\phi T(\beta_{s,T-1}^{\max}(\bullet)+\beta_{e,T-1}^{\max}(\bullet))<\infty.
\end{aligned}$$

Using the conditions in (21) and (22) with $\iota=\beta_\theta(T,\bullet)$, it follows that $|\mathcal{A}_k^e(T)|<1$ for any $k\in\mathbb{Z}_+$, indicating the bounded convergence of the output tracking error $e_k(T+1)$ along the batch direction, i.e.

$$\begin{aligned}
&\sup_{k\in\mathbb{Z}_+}|e_k(T+1)|\leq\beta_e(T+1,\bullet)<\infty,\\
&\limsup_{k\to\infty}|e_k(T+1)|\leq\beta_{e_{\sup}}(T+1,\bullet)<\beta_e(T+1,\bullet).
\end{aligned}$$

The boundedness of $e_k(T+1)$ together with the fact that $|y_{k-1}(i)|\leq\beta_{\mathrm{d}}+\beta_{e,T-1}^{\max}(\bullet)$ and $|y_{k-1}^s(i)|\leq\beta_{s,T-1}^{\max}(\bullet)$ for any $i\in\mathbb{Z}_{T-1}$ ensures that

$$\begin{aligned}
&|h_k(T)r_{2,k}(T)|\\
&\leq|\alpha|\beta_\theta(T)\Big\{\beta_e(T+1,\bullet)+|\tau|\beta_\phi[\beta_{\mathrm{d}}+\beta_e(T,\bullet)]\\
&\quad+|\tau_{\mathrm{I}}|\beta_\phi T(\beta_{s,T-1}^{\max}(\bullet)+\beta_{\mathrm{d}}+\beta_{e,T-1}^{\max}(\bullet))\\
&\quad+\beta_\phi T\left[|\tau|+(T-1)|\tau_{\mathrm{I}}|\right](\beta_{s,T-1}^{\max}(\bullet)+\beta_{\mathrm{d}}+\beta_{e,T-1}^{\max}(\bullet))\\
&\quad+2T\beta_\phi\beta_w+2\beta_w+2\beta_\phi\beta_\delta\Big\}<\infty.
\end{aligned}$$

Based on the conditions in (21) and (22) with $\iota=\beta_\theta(T,\bullet)$, it follows $|\mathcal{A}_k^s(T)|<1$ for any $k\in\mathbb{Z}_+$. Therefore, the bounded convergence of $y_k^s(T)$ is guaranteed and hence is assumed to satisfy $\sup_{k\in\mathbb{Z}_+}|y_k^s(T)|\leq\beta_s(T,\bullet)$, where $\beta_s(T,\bullet)>0$ is a finite constant related to $\beta_{\Delta\omega}$ and $\beta_{\Delta\delta}$.

By the mathematical induction, the conclusion in Theorem 2 follows immediately. Since the time duration is finite for the batch operation, a uniform threshold of $\iota=\max_{i\in\mathbb{Z}_T}\beta_\theta(i,\bullet)$ can be taken.This completes the proof.

Table 1: Parameter settings of different methods for Example 1

Parameters	Proposed	IDL-iAILC	OAILC
Feedback controller	$\tau_{\mathrm{P}} = 0.01$ $\tau_{\mathrm{I}} = 0.46$	$K = 0.15$	—
Learning controller	$\eta = 0.1$ $\mu = 0.6$ $\rho = 1$ $\lambda = 1$ $\alpha = 0.7$ $\varepsilon_1 = 1e-7$ $\varepsilon_1 = 1e-7$	$\eta = 0.1$ $\mu = 0.6$ $\eta_{nl} = 1$ $\mu_{nl} = 1$ $\alpha = 3.6$ $\varepsilon_2 = 1e-7$ $\varepsilon_3 = 1e-7$	$\lambda = 1$ $\mu_1 = 1$ $\mu_2 = 0.001$ $\gamma_1 = 0.8$ $\gamma_2 = 0.16$ $\gamma_3 = 0.04$ $\varepsilon = 0.01$
Initial conditions	$\hat{\phi}_0(t) = 1$ $\theta_0(t) = 1$	$\hat{\phi}_0(t) = 1$ $\theta_0(t) = 1$	$\hat{\theta}_0(t) = 1$

Remark 6 *The sufficient conditions in Theorem 2 ensure not only the bounded convergence of output tracking error, the bounded set-point command and adaptive learning gain, but also the bounded convergence of IESO along the batch direction. Moreover, the sufficient condition in* (40) *for the bounded convergence of IESO is independent of those in* (21) *and* (22)*, thus facilitating the parameter tuning of IESO.*

7 Illustrative examples

Two illustrative examples from the literature are used to verify and compare the new designs in this paper with the recently developed DDILC methods. In both examples, the process models are merely used to generate I/O data for control design.

Example 1 *Consider an unknown nonlinear time-varying system studied in [21, 29]*

$$y_k(t+1) = \begin{cases} \frac{y_k(t)}{1+y_k^2(t)} + u_k^3(t) + \omega_k(t), & t \in [0, 50], \\ \frac{y_k(t)y_k(t-1)y_k(t-2)u_k(t-1)[y_k(t-2)-1]+a(t)u_k(t)}{1+y_k^2(t-1)+y_k^2(t-2)} & \\ \quad +\omega_k(t), & t \in (50, 100], \end{cases}$$

where $a(t) = round(t/50) + 1$*,* $\omega_k(t) = 0.2sin(t/70 + k/50) + \delta_{1,k}(t)$ *and* $y_k(0) = -1 + \delta_k$*,* $\delta_{1,k}(t)$ *and* δ_k *are nonrepetitive and vary randomly within* $[-0.05, 0.05]$*. Note that the aforementioned Assumptions 2 and 3 are obviously satisfied owing to no abrupt changes in the system inputs and outputs. Besides, Assumption 4 is also satisfied under the above disturbances and initial conditions.*

The desired reference trajectory is the same as that in [29]

$$y_d(t+1) = \begin{cases} 0.5 \times (-1)^{round(t/10)}, & t \in [0, 30], \\ 0.5sin(t\pi/10) + 0.3cos(t\pi/10), & t \in (30, 70], \\ 0.5 \times (-1)^{round(t/10)}, & t \in (70, 100]. \end{cases}$$

In the case when $\lambda_t = 0.15$*,* $\rho_t = 0.75$*,* $\eta_t = 1$*,* $\mu_t = 0.2$ *and* $\varepsilon_t = 10^{-8}$ *in* (9) *and* (11) *for validation, the tuning regions of the PI controller parameters are determined as* $\tau_P \in [0.01, 0.87]$

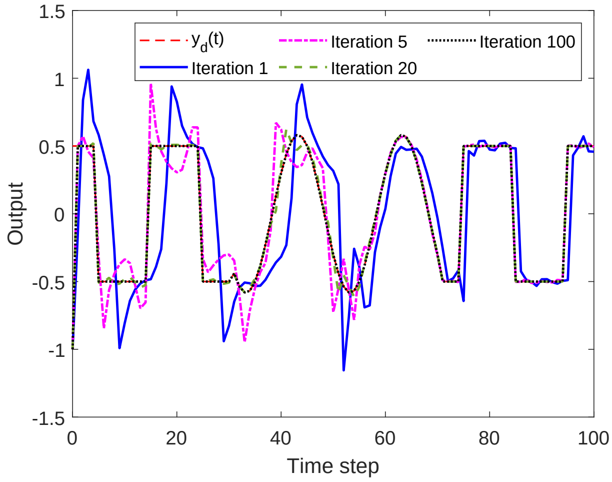

Figure 3: Tracking results by the new design applied to Example 1 in the absence of external disturbances and initial shifts.

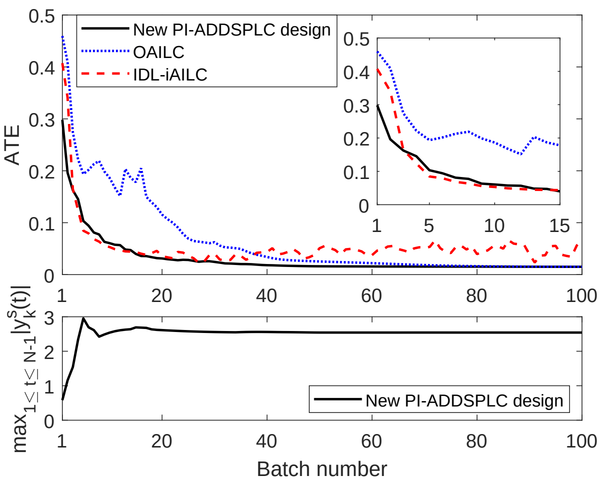

Figure 4: Plot of ATE indices by different methods and the set-point command assessed by $\max_{1\leq t\leq N-1}|y_k^s(t)|$ for the new design applied to Example 1 in the absence of external disturbances and initial shifts.

and $\tau_I \in [0.30, 0.97]$*. Applying Algorithm 1 with* $\varrho = 0.3$ *and* $\Delta\tau_P = \Delta\tau_I = 0.01$*, a set of desired PI controller parameters,* $\tau_P = 0.01$ *and* $\tau_I = 0.46$*, can then be obtained. The parameter settings of the proposed PI-ADDSPLC scheme are listed in Table 1. Similarly, the recently developed iterative dynamic linearization-based indirect adaptive ILC (IDL-iAILC) in [29] and the optimization-based adaptive ILC (OAILC) [25] are also applied for comparison, for which the corresponding parameter settings are also listed in Table 1. Note that the iAILC method given in [29] cannot be applied to nonlinear processes and therefore is not considered herein.*

Figs. 3 and 4 show the tracking results and the corresponding ATE indices together with the set-point command assessed by $\max_{1\leq t\leq N-1}|y_k^s(t)|$*, respectively, in the absence of external disturbances and initial shifts. It is seen that perfect tracking is almost realized after 20 batches by the new PI-ADDSPLC scheme. In contrast, nearly 80 batches are required to achieve the similar tracking*

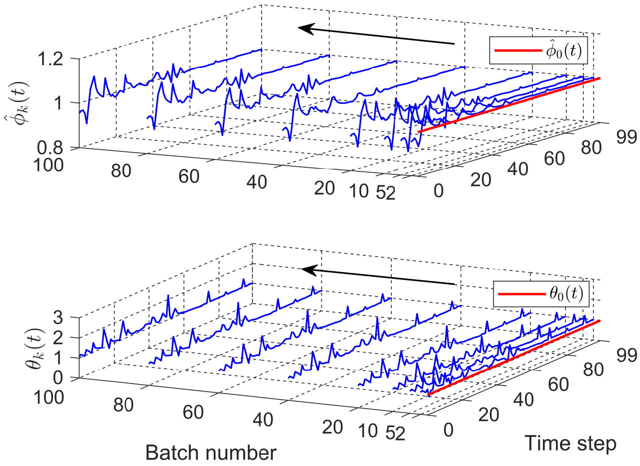

Figure 5: Evolutions of the derivative estimation $\hat{\phi}_{k,k-1}(t)$ and adaptive learning gain $\theta_k(t)$ by the new design applied to Example 1 in the absence of external disturbances and initial shifts.

precision by the OAILC design in [25], and the IDL-iAILC method in [29] leads to relatively larger steady-state tracking error along the batch direction. In addition, the evolutions of the derivative estimation $\hat{\phi}_k(t)$ and the adaptive learning gain $\theta_k(t)$ shown in Fig. 5 demonstrate their boundedness along both the time and batch directions, which may be used to roughly estimate a feasible region of the tuning parameter α as $(0, 1.75)$. Therefore, it is reasonable to select $\alpha = 0.7$ for implementation.

In the presence of nonrepetitive disturbance $\omega_k(t)$ and initial shifts $y_k(0)$, the new IESO based PI-ADDSPLC design is also evaluated, where the IESO gains are taken as $l_1 = 0.4$, $l_2 = 0.03$ to satisfy the premise in (40), a constraint on $\Delta y_k^s(t)$ is set as $\delta_{sp} = 10$, and the remaining parameters are taken as those in the nominal case. The resulting ATE indices and the set-point command assessed by $\max_{1 \le t \le N-1} |y_k^s(t)|$ are shown in Fig. 6. It is seen that both the tracking speed and steady-state tracking error by the proposed two new designs are evidently better than those by OAILC in [25] and IDL-iAILC [29]. Moreover, further improved tracking performance is obtained by the new IESO based PI-ADDSPLC design compared to PI-ADDSPLC.

Example 2 *Consider an industrial injection molding process studied in [8, 17, 18, 25, 35], for which a model used to generate the process data is*

$$\begin{aligned} y_k(t+1) = 1.607 y_k(t) - 0.6086 y_k(t-1) \\ + 1.239 u_k(t) - 0.9282 u_k(t-1) + \omega_k(t), \end{aligned}$$

where $\omega_k(t) = 10\, sin(t/10 + k/30) + \delta_{1,k}(t)$ and the initial process output is taken as $y_k(0) = \delta_{2,k}$, $\delta_{1,k}(t)$ and $\delta_{2,k}$ represent uncertain parameters randomly varying in the interval $[-1, 1]$. It is easy to verify that the Assumptions 2-4 are satisfied in this example. The desired reference trajectory is

$$y_d(t) = \begin{cases} 150, & 0 \le t \le 50, \\ 300, & 51 \le t \le 100. \end{cases}$$

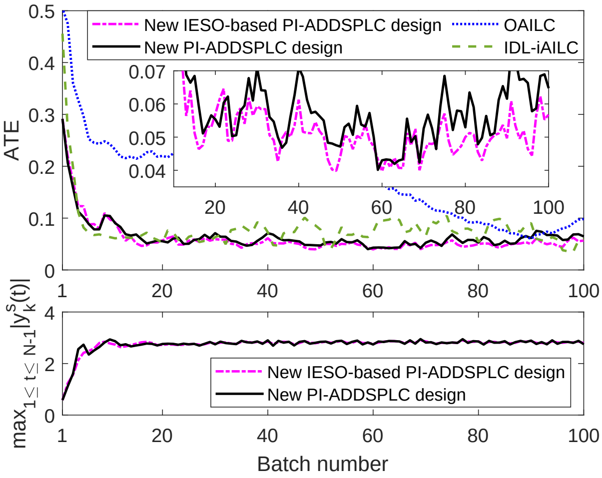

Figure 6: Plot of ATE indices by different methods and the set-point command assessed by $\max_{1\leq t\leq N-1}|y_k^s(t)|$ for the new designs applied to Example 1 with nonrepetitive uncertainties.

Table 2: Parameter settings of different methods for Example 2

Parameters	Proposed	iAILC	IDL-iAILC	OAILC
FC	$\tau_{\rm P}=0.44$ $\tau_{\rm I}=0.5$	$K=1$	$K=1$	—
LC	$\eta=0.01$ $\mu=2$ $\rho=0.1$ $\lambda=1$ $\alpha=0.1$ $\varepsilon_1=1e{-}7$ $\varepsilon_1=1e{-}7$	$\eta_l=0.01$ $\mu_l=2$ $\alpha=0.1$ $\varepsilon_1=1e{-}7$	$\eta=0.01$ $\mu=2$ $\eta_{nl}=0.1$ $\mu_{nl}=1$ $\alpha=0.1$ $\varepsilon_2=1e{-}7$ $\varepsilon_3=1e{-}7$	$\lambda=1$ $\mu_1=1$ $\mu_2=0.001$ $\gamma_1=0.95$ $\gamma_2=0.05$ $\varepsilon=1e{-}3$
IC	$\hat{\phi}_0(t)=1$ $\theta_0(t)=1$	$\theta_0(t)=1$	$\hat{\phi}_0(t)=1$ $\theta_0(t)=1$	$\hat{\theta}_0(t)=1$

* FC, LC and IC represent feedback controller, learning controller and initial conditions, respectively.

In the case when $\lambda_t=1$, $\rho_t=1$, $\eta_t=1$, $\mu_t=1$ *and* $\varepsilon_t=10^{-8}$ *are taken in* (9) *and* (11) *for validation, the tuning regions of PI controller parameters are determined as* $\tau_P\in[0.14,0.5]$ *and* $\tau_I\in[0.48,0.5]$. *Note that there are two groups of steady-state values of the PI controller parameters, i.e.,* $\tau_P=0.34,\tau_I=0.48$ *and* $\tau_P=0.44,\tau_I=0.5$, *due to the piecewise function structure of the desired reference trajectory. By comparing the corresponding ATE indices when applying these two groups of PI controller parameters,* $\tau_P=0.44$ *and* $\tau_I=0.5$ *are used for the set-point learning scheme in this example.*

For clarity, the parameter settings in the new PI-ADDSPLC scheme are listed in Table 2, where the PI controller parameters indicate their steady-state values. For comparison, the recently developed indirect adaptive ILC (iAILC) and IDL-iAILC designs in [29] along with OAILC in [25] are also performed. The corresponding parameter settings for these data-driven control methods in *terms of the guidelines given therein are also listed in Table 2.*

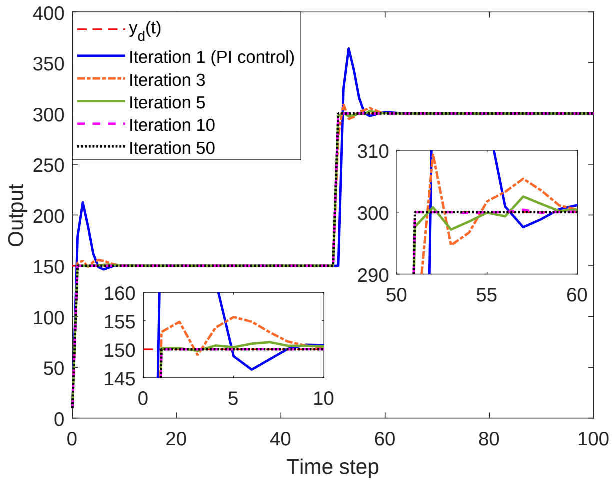

Figure 7: Tracking results by the new design applied to Example 2 in the absence of external disturbances and initial shifts.

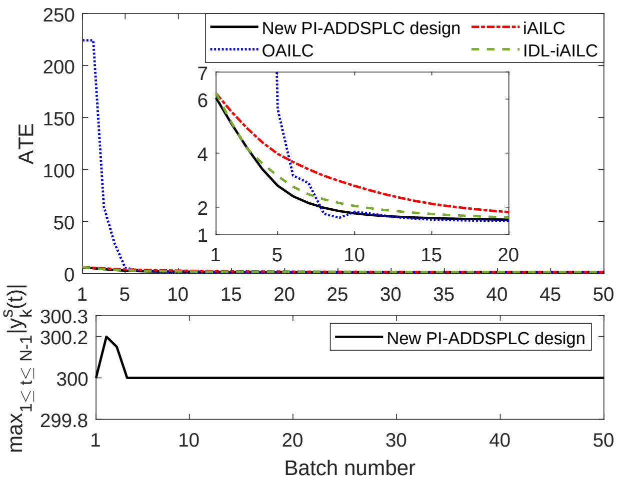

Figure 8: Plot of ATE indices by different methods and the set-point command assessed by $\max_{1\le t\le N-1}|y_k^s(t)|$ for the new design applied to Example 2 in the absence of external disturbances and initial shifts.

In the absence of external disturbances and initial shifts, i.e., $\omega_k(t) = 0$ *and* $y_k(0) = 0$*, the tracking results are shown in Fig. 7. Correspondingly, the ATE indices by different methods and the set-point command assessed by* $\max_{1\le t\le N-1}|y_k^s(t)|$ *are plotted in Fig. 8. It is seen that perfect tracking can be achieved after 10 batches by the new design, while the tracking speed is obviously faster than those of iAILC and IDL-iAILC [29] and OAILC [25]. Moreover, the bounded set-point command is guaranteed, as shown in the bottom plot of Fig. 8.*

Note that the tracking error in the initial batch run by the new design is significantly reduced in contrast to the existing iAILC and IDL-iAILC methods in [29] and OAILC method in [25]. Meanwhile, the evolutions of the derivative estimation $\hat{\phi}_k(t)$ *and the learning gain* $\theta_k(t)$ *shown in*

Fig. 9 well demonstrate their boundedness along both the time and batch directions. A feasible region of the parameter α is approximately determined as $(0, 0.16)$ based on the simulation bounds of $\hat{\phi}_k(t)$ and $\theta_k(t)$, verifying the appropriateness of choosing $\alpha = 0.1$ in the new design.

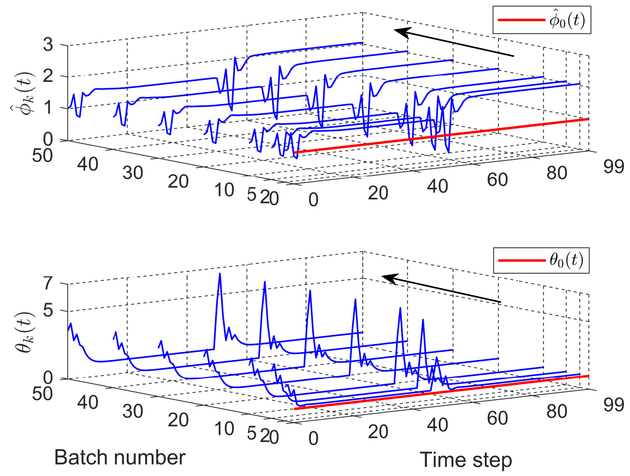

Figure 9: Evolutions of the derivative estimation $\hat{\phi}_{k,k-1}(t)$ and adaptive learning gain $\theta_k(t)$ by the new design applied to Example 2 in the absence of external disturbances and initial shifts.

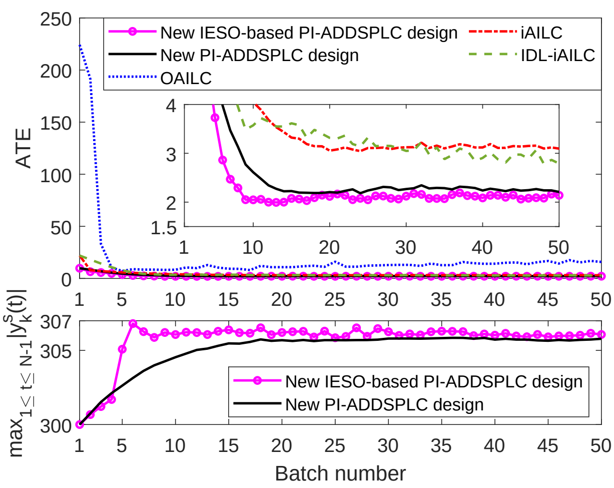

Figure 10: Plot of ATE indices by different methods and the set-point command assessed by $\max_{1\leq t\leq N-1} |y_k^s(t)|$ for the new designs applied to Example 2 with nonrepetitive uncertainties.

In the presence of nonrepetitive initial conditions and disturbances, the new IESO based PI-ADDSPLC design is also applied for comparison, where the IESO gains are taken as $l_1 = 1$ and $l_2 = 0.8$ to satisfy the premise in (40). Meanwhile, a constraint on the batch-wise variation of set-point command is set as $\delta_{\rm sp} = 50$. The corresponding ATE indices and the set-point command assessed by $\max_{1\leq t\leq N-1} |y_k^s(t)|$ are shown in Fig. 10. It is seen that the two new designs well maintain the robust convergence of the resulting ILC system. Also, the tracking performance is significantly improved compared with the recently developed methods in [25] and [29]. Note that the *new IESO-based PI-ADDSPLC outperforms PI-ADDSPLC owing to the disturbance compensation. Moreover, the real-time updated set-point command is bounded, as per Theorem 2.*

8 Conclusions

For linear/nonlinear batch processes with unknown dynamics subject to nonrepetitive initial conditions and disturbances, this paper has developed a robust PI-ADDSPLC scheme for batch run optimization, based on a widely used PI control loop by only using the process I/O data. Significantly enhanced performance is achieved from the initial batch compared to existing DDILC methods based on the P-type feedback structure, such as the recently developed iAILC and IDL-iAILC [29]. Robust convergence of the output tracking error together with the boundedness of adaptive learning gain and real-time updated set-point command has been analyzed using the mathematical induction. Moreover, by introducing an IESO into the PI-ADDSPLC, another set-point learning control scheme of IESO-based PI-ADDSPLC has been developed to further enhance the tracking performance in the presence of nonrepetitive uncertainties. A nonlinear numerical example and an industrial injection molding process have been used to demonstrate the effectiveness and superiority of the proposed two new ADDSPLC schemes. It should be noted that the new ADDSPLC schemes are based on the classical PI control loop to construct a set-point learning system. The extension to more advanced closed-loop control structures deserves further exploration, in particular for nonlinear batch processes with input or output delay.

References

[1] F. Gao, Y. Yang, and C. Shao, "Robust iterative learning control with applications to injection molding process," *Chem. Eng. Sci.*, vol. 56, no. 24, pp. 7025–7034, 2001.

[2] Z. K. Nagy, "Model based robust control approach for batch crystallization product design," *Comput. Chem. Eng.*, vol. 33, no. 10, pp. 1685–1691, Oct. 2009.

[3] H.-S. Ahn, Y.-Q. Chen, and K. Moore, "Iterative learning control: Brief survey and categorization," *IEEE Trans. Syst., Man, Cybern., C, Appl. Rev.*, vol. 37, no. 6, pp. 1099–1121, Nov. 2007.

[4] Y. Wang, F. Gao, and F. Doyle, "Survey on iterative learning control, repetitive control, and run-to-run control," *J. Process Control*, vol. 19, no. 10, pp. 1589–1600, Dec. 2009.

[5] D. Shen, "Iterative learning control with incomplete information: A survey," *IEEE/CAA J. Autom. Sinica*, vol. 5, no. 5, pp. 885–901, Sep. 2018.

[6] S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering operation of robots by learning," *J. Robot Syst.*, vol. 1, no. 2, pp. 123–140, 1984.

[7] 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 Syst.*, vol. 26, no. 3, pp. 96–114, Jun. 2006.

[8] J. Shi, F. Gao, and T.-J. Wu, "Robust design of integrated feedback and iterative learning control of a batch process based on a 2D Roesser system," *J. Process Control*, vol. 15, no. 8, pp. 907–924, Dec. 2005.

[9] J. Bolder and T. Oomen,"Inferential iterative learning control: a 2D-system approach", *Automatica*, vol.71, pp. 247–253, Sep. 2016.

[10] D. Meng and K.L. Moore, "Convergence of iterative learning control for SISO nonrepetitive systems subject to iteration-dependent uncertainties," *Automatica*, 79, pp. 167–177, May 2017.

[11] S. V. Johansen, M. R. Jensen, B. Chu, J. D. Bendtsen, J. Mogensen, and E. Rogers, "Broiler FCR optimization using norm optimal terminal iterative learning control", *IEEE Trans. Control Syst. Technol.*, vol. 29, no. 2, pp. 580–592, Mar. 2021.

[12] D. Meng and K. L. Moore, "Robust iterative learning control for nonrepetitive uncertain systems," *IEEE Trans. Autom. Control*, vol. 62, no. 2, pp. 907–913, Feb. 2017.

[13] D. Meng and J. Zhang, "Convergence analysis of robust iterative learning control against nonrepetitive uncertainties: system equivalence transformation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 9, pp. 3867–3879, Sep. 2021.

[14] Y. Chen, C. Wen and M. Sun," A robust high-order P-type iterative learning controller using current iteration tracking error," *Int J. Control*, 1997, vol. 68, no. 2, pp. 331–342.

[15] S. Mandra, K. Galkowski, E. Rogers, A. Rauh, and H. Aschemann, "Performance-enhanced robust iterative learning control with experimental application to PMSM position tracking," *IEEE Trans. Control Syst. Technol.*, vol. 27, no. 4, pp. 1813–1819, Jul. 2019.

[16] S. Hao, T. Liu, W. Paszke, K. Galkowski, and Q.-G. Wang, "Robust static output feedback based iterative learning control design with a finite-frequency-range two-dimensional $\mathcal{H}_\infty$ specification for batch processes subject to nonrepetitive disturbances," *Int. J. Robust Nonlinear Control*, vol. 31, no. 12, pp. 5745–5761, Aug. 2021.

[17] T. Liu, X. Wang, and J. Chen, "Robust PID based indirect-type iterative learning control for batch processes with time-varying uncertainties," *J. Process Control*, vol. 24, no. 12, pp. 95–106, Dec. 2014.

[18] S. Hao, T. Liu, and F. Gao, "PI based indirect-type iterative learning control for batch processes with time-varying uncertainties: A 2D FM model based approach," *J. Process Control*, vol. 79, pp. 57–67, Jun. 2019.

[19] J. Shi, H. Zhou, Z. Cao, and Q. Jiang, "A design method for indirect iterative learning control based on two-dimensional generalized predictive control algorithm," *J. Process Control*, vol. 24, no. 10, pp. 1527–1537, Oct. 2014.

[20] S. Hao, T. Liu, and E. Rogers, "Extended state observer based indirect-type ILC for single-input single-output batch processes with time- and batch-varying uncertainties," *Automatica*, 112, 108673, Feb. 2020.

[21] Z. Hou and S. Jin, *Model Free Adaptive Control: Theory and Applications*. New York, NY, USA: CPC Press, 2013.

[22] R. Chi, Y. Hui, and Z. Hou, *Data-Driven Iterative Learning Control for Discrete-Time Systems*. Springer Nature, 2022.

[23] Z. Hou, R. Chi, and H. Gao, "An overview of dynamic-linearization-based data-driven control and applications," *IEEE Trans. Ind. Electron.*, vol. 64, no. 5, pp. 4076–4090, May 2017.

[24] M. Benosman, "Model-based vs data-driven adaptive control: An overview. *Int. J. Adap. Control Signal Process.*, vol. 32, no. 1, pp. 1–24. May 2018.

[25] D. Meng and J. Zhang, "Design and analysis of data-driven learning control: An optimization-based approach," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 10, pp. 5527–5541, Oct. 2022.

[26] Q. Yu, Z. Hou, X. Bu, and Q. Yu, "RBFNN-based data-driven predictive iterative learning control for nonaffine nonlinear systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 4, pp. 1170–1182, Apr. 2020.

[27] X. Yu, Z. Hou, and M. M. Polycarpou, "Controller-dynamic-linearization-based data-driven ILC for nonlinear discrete-time systems with RBFNN," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 52, no. 8, pp. 4981–4992, Aug. 2022.

[28] R. Chi, Z. Hou, S. Jin, and B. Huang, "Computationally efficient data-driven high order optimal iterative learning control," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 12, pp. 5971–5980, Dec. 2018.

[29] R. Chi, H. Li, D. Shen, Z. Hou, and B. Huang, "Enhanced P-type control: indirect adaptive learning from set-point updates," *IEEE Trans. Autom. Control*, vol. 68, no. 3, pp. 1600–1613, Mar. 2023.

[30] K. J. Åström and T. Hägglund, "PID controllers: Theory, design and tuning. In *Research triangle park*. NC: Instrument society of America, 1995.

[31] C. Zhao and L. Guo, "Towards a theoretical foundation of PID control for uncertain nonlinear systems," *Automatica*, vol. 142, 110360, Aug. 2022.

[32] H. Yu, Z. Guan, T. Chen, and T. Yamamoto, "Design of data-driven PID controllers with adaptive updating rules," *Automatica*, vol. 121, 109185, Nov. 2020.

[33] S. Xiong and Z. Hou, "Stabilizing regions of PID controller for a class of unknown nonlinear non-affine discrete-time systems," *Int. J. Robust Nonlinear Control*, vol. 32, no. 18, pp. 9421–9437, Dec. 2022.

[34] R. Chi, Y. Hui, B. Huang, and Z. Hou, "Active disturbance rejection control for nonaffined globally Lipschitz nonlinear discrete-time systems," *IEEE Trans. Autom. Control*, vol. 66, no. 12, pp. 5955–5967, Dec. 2021.

[35] D. Meng and J. Zhang, "Robust optimization-based iterative learning control for nonlinear systems with nonrepetitive uncertainties," *IEEE/CAA J. Automatica Sinica*, vol. 8, no. 5, pp. 1001–1014, May 2021.

[36] Z. Hou and S. Xiong, "On model-free adaptive control and its stability analysis," *IEEE Trans. Autom. Control*, vol. 64, no. 11, pp. 4555–4569, Nov. 2019.

[37] R. Chi, Z. Hou, B. Huang, and S. Jin, "A unified data-driven design framework of optimality-based generalized iterative learning control," *Comput. Chem. Eng.*, vol.77, pp. 10–23, Jun. 2015.

[38] J. Zhang and D. Meng, "Improving tracking accuracy for repetitive learning systems by high-order extended state observers," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 12, pp. 10398–10407, Dec. 2023.

[39] S. Hao, T. Liu, and W. Paszke, "PIO based data-driven iterative learning control for nonlinear batch processes with nonrepetitive disturbances subject to input constraints," *IFAC PapersOnLine*, vol. 54, no. 3, pp. 25–30, Jun. 2021.

[] Shoulin Hao (Member, IEEE) received his Ph.D degree in Control Theory and Control Engineering from Dalian University of Technology, Dalian, China, in 2018. He is an associate professor with the Institute of Advanced Control Technology, Dalian University of Technology, Dalian, China. His research interests include industrial process control, iterative learning control and data-driven control. He published more than 40 research papers and coauthored one monograph.

[] Tao Liu (Senior Member, IEEE) received his Ph.D degree in control science and engineering from Shanghai Jiaotong University, Shanghai, China, in 2006. He is a professor with the Institute of Advanced Control Technology, Dalian University of Technology, Dalian, China. His research interests include industrial process measurement, modeling & control, batch process optimization, data-driven control. He published more than 150 research papers and two monographs. He serves as an associate editor of IEEE Transactions on Industrial Informatics, editorial board member of International Journal of Control.

[]Eric Rogers (Life Member, IEEE) is Professor of Control Systems Theory and Design, School of Electronics and Computer Science, University of Southampton, UK. His major research interests are in iterative learning control theory and applications in both engineering and healthcare, multidimensional systems theory and applications, long range autonomy for autonomous underwater vehicles and flow control. He is the Editor-in-Chief of the International Journal of Control and Multidimensional Systems and Signal Processing, and has an extensive record of advisory work for funding agencies and industries in the UK and elsewhere.

[] Youqing Wang (Senior Member, IEEE) received his Ph.D degree in Control Science and Engineering from Tsinghua University, Beijing, China, in 2008. He worked chronologically at Hong Kong University of Science and Technology, Hong Kong, China; University of California, Santa Barbara, CA, USA; Shandong University of Science and Technology, Qingdao, China. He is currently a Professor at Beijing University of Chemical Technology, Beijing, China. His research interests include fault diagnosis, fault-tolerant control, state monitoring and iterative learning control for chemical and biomedical processes. He was the recipient of IET Fellow in 2022, NSFC Distinguished Young Scientists Fund in 2022.

[]Wojciech Paszke (Member, IEEE) received his M.Sc. and Ph.D degrees in Electrical Engineering from the Technical University of Zielona Góra, Góra, Poland, in 2000 and 2005, respectively. From 2008 to 2010, he was affiliated to Eindhoven University of Technology, Eindhoven, The Netherlands, where he has been a control systems expert on high precision positioning of electron microscope. He is currently affiliated to the Institute of Automation, Electronic and Electrical Engineering, University of Zielona Góra. His research is focused on multidimensional (nD) systems, repetitive processes, iterative learning control schemes, and convex optimization in robust control problems.