IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received June 18, 2019, accepted July 27, 2019, date of publication August 12, 2019, date of current version August 28, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2934474

Quadratic-Criterion-Based Model Predictive Iterative Learning Control for Batch Processes Using Just-in-Time-Learning Method

LIUMING ZHOU, LI JIA, AND YU-LONG WANG, (Member, IEEE)

Shanghai Key Laboratory of Power Station Automation Technology, Department of Automation, School of Mechatronic Engineering and Automation, Shanghai University, Shanghai 200072, China

Corresponding author: Li Jia (jiali@staff.shu.edu.cn)

This work was supported in part by the National Science Foundation of China under Grant 61773251, Grant 61873335, and Grant 61833011, and in part by the Program for Professor of Special Appointment (Eastern Scholar) at Shanghai Institutions of Higher Learning, China.

ABSTRACT In this paper, a new quadratic-criterion-based model predictive iterative learning control (QMPILC) algorithm for tracking problem of batch processes is proposed. In the proposed QMPILC design, a parametric time-varying model consisting of a set of local models is established for nonlinear batch processes by using the just-in-time-learning method. In order to describe the processes more accurately, the model is updated with batch running. On basis of the identification model, iterative learning control is combined with model predictive control based on a quadratic performance criterion, and the control law can be obtained by solving a convex optimization problem. According to the real-time feedback information, the input is updated to reject real-time disturbance. As a result, the proposed QMPILC algorithm improves control performance and optimization efficiency. In addition, the convergence and tracking performance of QMPILC are analyzed. The proposed methods are illustrated on batch reactor. The results are provided to show excellent performance of tracking product qualities.

INDEX TERMS Iterative learning control, batch processes, just-in-time-learning, local models, model predictive control.

I. INTRODUCTION

Iterative learning control (ILC) is an effective control technique for systems which have a repeat movement characteristic. The general idea of ILC is to update control signal for the current batch by using the information of the pervious batches. Then output trajectory converges to desired reference trajectory after several iterations [1]–[3]. Thus, ILC is suitable for the control systems whose control task is tracking desired trajectory and ends in a finite time, and it has been widely applied in batch processes whose characteristics are repetitive, nonlinear and time-varying [4], [5]. Batch processes systems are one of the most important research areas in process industry [6], and they have been widely applied to the manufacture of low-volume and high-value products such as semiconductors, pharmaceuticals, polymeric materials, and injection products. [7].

In many general ILC algorithms, the control input of the current batch is calculated before the beginning of the batch [8]–[11]. However, it is difficult to guarantee the control performance when there exists real-time disturbance since the type of ILC is an open-loop control. At present, many researches focus on studying the convergence [12], [13] and the ability of reject disturbance of ILC [14]. In order to deal with real-time disturbance and uncertainties, it is reasonable to combine ILC with other real-time feedback algorithms such as PID [15] and MPC [16]. Then the inputs of the systems can be adjusted according to the feedback information. Oh *et al.* proposed a MPC technique combined with ILC to reject real-time disturbance [17]. Lu *et al.* proposed a new two-stage design based on the combination of ILC and MPC [18]. Shi *et al.* presented a two-dimensional control scheme by combining ILC in the outer loop and MPC in the inner loop [19]. Wang *et al.* proposed an advanced ILC-based PI control for batch processes [20]. In above mentioned results, the controllers are designed based on linear

The associate editor coordinating the review of this article and approving it for publication was YangQuan Chen.

time-invariant models. However, the linear time-invariant models usually can not describe the nonlinearity of batch process accurately. Thus, it may lead to worse control performance for nonlinear batch processes due to serious model errors. ILC strategies for nonlinear process have also been studied in some papers. Lee *et al.* presented a batch-MPC based on a linearized model [21]. Nevertheless, establishing a mechanism model is often difficult and costly. Jia *et al.* proposed a quadratic criterion-ILC scheme based on data-driven models [22], [23]. The control input was obtained by using some intelligent optimization algorithms such SQP, PSO and GA, which may lead to a suboptimal solution for the optimization problem. In addition, these optimization algorithms are often complex and time-consuming. Su *et al.* proposed an integrated B2B-NMPC control method based on a multiway partial least-squares models [24]. However, a rigorous convergence analysis to verify the control strategy was not presented. Therefore, the aim of this paper is to develop an integrated control scheme which can improve the real-time control performance and the optimization efficiency of the system for batch processes with strong nonlinearity. In this paper, considering the problems of nonlinearity, constraints and real-time control, we propose a quadratic-criterion-based model predictive iterative learning control (QMPILC) algorithm by using the just-in-time-learning (JITL) method. A time-varying parameter model is established by linearizing the nonlinear model along the nominal trajectories according to input and output data. Then, the controller is designed based on a quadratic performance criterion, and the input of the system is obtained by solving a convex optimization problem.

The main contributions of this paper are summarized as follows:

(1) Based on the JITL method, the design of the model predictive iterative learning controller for nonlinear batch processes is proposed. The control law is obtained by solving a convex optimization problem, which improves the efficiency of the system.

(2) The analyses of the convergence and the tracking performance for the proposed system are given in this paper.

The rest of the paper is organized as follows: In Section 2, a data-driven based model is established for nonlinear batch processes. In Section 3, a design method of model predictive iterative controller is presented. The convergence and stability results are given in Section 4. Section 5 shows the results of applying the proposed method to batch processes, and some conclusions are drawn in Section 6.

II. BATCH PROCESS DESCRIPTION

In this paper, a single-input single-output process is considered. The batch run length is fixed, and it is divided into N equal intervals. The model for nonlinear batch processes can be described as follows:

$$\begin{aligned} y_k(t) = f(&y_k(t-1), y_k(t-2), \cdots, y_k(t-n_y), u_k(t-1), \\ &u_k(t-2), \cdots, u_k(t-n_u)) \\ &t = 1, \cdots, N; \quad k = 1, 2, \cdots \end{aligned} \tag{1}$$

where t is the discrete-time index; subscript k is the batch index; $y \in R$ and $u \in R$ denote the product quality variable (output) and the control variable (input), respectively; n_y and n_u are integers related to the model order; and f is the nonlinear function.

The JITL method is a data-based methodology for nonlinear process modeling. It can be used to approximate a nonlinear processes by establishing local models in the operating range of interest. According to query data, relevant data samples are selected from reference database based on some similarity criterion. Then local models are established based on the relevant data, and a process model is constructed by using the local models.

The nominal input trajectory and the corresponding output trajectory are defined as

$$U_k^s = [u_k^s(0), u_k^s(1), \cdots, u_k^s(N-1)]^T \tag{2}$$

$$Y_k^s = [y_k^s(1), y_k^s(2), \cdots, y_k^s(N)]^T \tag{3}$$

We can obtain the following local linear deviation model by subtracting the nominal trajectories from the operation trajectories.

$$\bar{y}_k(t) = \frac{B_k(q,t)}{A_k(q,t)}\bar{u}_k(t) \tag{4}$$

where q is the time-wise unit forward-shift operator, $\bar{y}_k(t) = y_k(t) - y_k^s(t)$, $\bar{u}_k(t) = u_k(t) - u_k^s(t)$, $A_k(q,t) = 1 + a_k^1(t)q^{-1} + a_k^2(t)q^{-2} + \cdots + a_k^{n_y}(t)q^{-n_y}$, and $B_k(q,t) = b_k^1(t)q^{-1} + b_k^2(t)q^{-2} + \cdots + b_k^{n_u}(t)q^{-n_u}$. Here, $a_k^i(t)$ and $b_k^i(t)$ are coefficients which vary with time and batch. Since the JITL method usually employs a second-order model [25], the following local model is considered.

$$\hat{\bar{y}}_k(t) = a_k^1(t)\bar{y}_k(t-1) + a_k^2(t)\bar{y}_k(t-2) + b_k(t)\bar{u}_k(t-1) \tag{5}$$

Then a database $\{\Xi_(t), \Phi(t)\}$ at time t is constructed based on the historical process data as

$$\Xi_(t) = [y_1(t), y_2(t), \cdots, y_n(t)]^T \tag{6}$$

$$\Phi(t) = \begin{bmatrix} y_1(t-1) & y_1(t-2) & u_1(t-1) \\ y_2(t-1) & y_2(t-2) & u_2(t-1) \\ \vdots & \vdots & \vdots \\ y_n(t-1) & y_n(t-2) & u_n(t-1) \end{bmatrix} \tag{7}$$

where n is the number of the process data in database.

For query data $q_k(t) = [y_k^s(t-1), y_k^s(t-2), u_k^s(t-1)]$ obtained from the nominal trajectory, a relevant dataset $\{\Xi_L(t), \Phi_L(t)\}$ consists of L samples which are select from $\{\Xi_(t), \Phi(t)\}$ based on a similarity criterion [26]. By subtracting $y_k^s(t)$ and $q_k(t)$ from the relevant dataset, we can obtain a relevant deviation dataset $\{\bar{\Xi}_L(t), \bar{\Phi}_L(t)\}$. Then the coefficients of the local model can be identified by using the least square method.

$$[a_k^1(t), a_k^2(t), b_k(t)]^T = (\bar{\Phi}_L^T(t)\bar{\Phi}_L(t))^{-1}\bar{\Phi}_L^T(t)\bar{\Xi}_L(t) \tag{8}$$

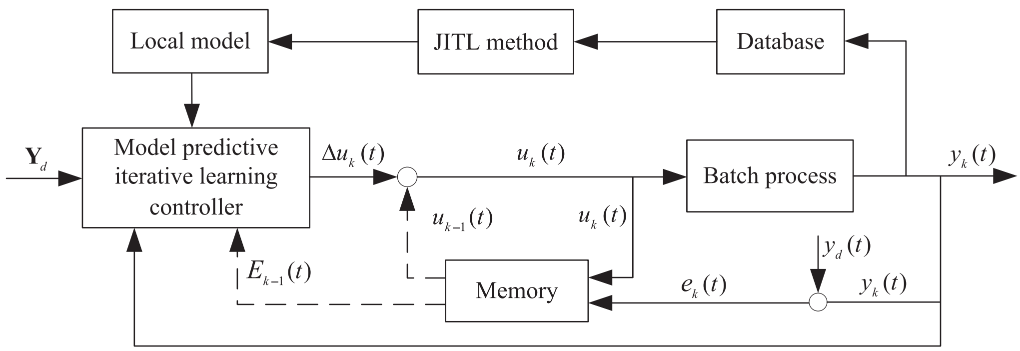


FIGURE 1. The structure of the QMPILC system.

A time-varying parameter state-space description is given as

$$x_k(t+1) = A_k(t+1)x_k(t) + B_k(t+1)\bar{u}_k(t)$$
$$\hat{\bar{y}}_k(t) = C_k x_k(t) \tag{9}$$

where

$$A_k(t) = \begin{bmatrix} 0 & 1 \\ a_k^2(t) & a_k^1(t) \end{bmatrix}, \quad B_k(t) = \begin{bmatrix} 0 \\ b_k(t) \end{bmatrix},$$
$$C_k = \begin{bmatrix} 0 \\ 1 \end{bmatrix}^T, \quad x_k(t) = \begin{bmatrix} x_k^1(t) \\ x_k^2(t) \end{bmatrix}$$

Denote

$$\hat{\bar{Y}}_k = [\hat{\bar{y}}_k(1), \hat{\bar{y}}_k(2), \ldots, \hat{\bar{y}}_k(N)]^T \tag{10}$$
$$\bar{U}_k = [\bar{u}_k(0), \bar{u}_k(1), \ldots, \bar{u}_k(N-1)]^T \tag{11}$$

According to Eq. (9), the system deviation model can be written as

$$\hat{\bar{Y}}_k = \hat{G}_k \bar{U}_k \tag{12}$$

where $\hat{G}_k$ is a matrix whose elements are $\hat{g}_k^{i,j} = C_k \prod_{l=0}^{i-j-1} A_k(i-l)B_k(j)(i,j = 1, 2, \cdots N)$.

The assumptions for the system are given as follows:

A1. $\hat{G}_k$ has a full row-rank;

A2. The initial state of the system is $x_k(0) = 0$.

The nonlinear processes are operated within a narrow range of the nominal trajectories, and they can be described by the time-varying parameter model which consists of a set of local linear models. The time-varying parameter model is recomputed after completing a batch run. Therefore, the model can well represent nonlinear batch processes.

III. DESIGN OF QMPILC STRATEGY FOR BATCH PROCESSES

The structure of the proposed QMPILC system is shown in Fig. 1. The model predictive iterative learning controller can perform real-time control based on the real-time feedback information and the tracking error of pervious batch. Let $Y_d = [y_d(1), y_d(2), \cdots, y_d(N)]^T$ represents the desired reference trajectory. The tracking error sequence of the process and of the deviation model are defined as $E_k = Y_d - Y_k$ and $\hat{E}_k = Y_d - \hat{Y}_k = \bar{Y}_{k,d} - \hat{\bar{Y}}_k$, respectively, where $\bar{Y}_{k,d} = Y_d - Y_k^s$ is the deviated reference trajectory in the kth batch. In this paper, the operation trajectory of the last batch is set as the nominal trajectory of the current batch. Both prediction horizon and control horizon in the QMPILC algorithm are N_p which decrease with time, namely, $N_p = N - t$. Then, at time t, we can obtain a predictive control sequence by solving a quadratic optimization problem, and only first element of the predictive control sequence is sent to process, namely, $u(t) = u(t|t)$.

Model accuracy and convergence rate is closely related to the trajectory of the initial batch. In this study, the initial input $u_1(t)$ can be obtained by using the local inverse model of the system as

$$u_1(t) = f^{-1}(y_d(t+1), y_1(t), y_1(t-1))$$
$$= \Theta_{inv}[y_d(t+1), y_1(t), y_1(t-1)]^T \tag{13}$$

where $\Theta_{inv} = [\alpha_{inv}^1(t), \alpha_{inv}^2(t), \beta_{inv}(t)]$ is the regression vector. By using the JITL method, the local inverse model can be established based on the data points near the query vector $q_0(t) = [y_d(t+1), y_1(t), y_1(t-1)]$ [27]. Then initial input $u_1(t)$ is computed according to the regression vector.

The following definition is given to represent the prediction horizon.

$$F_k(_{t_2}^{t_1}|t) = [f_k(t_1|t), f_k(t_1+1|t), \cdots, f_k(t_2|t)]^T$$

The MPC strategy can be used to improve the ability of rejecting disturbance. The output prediction $\hat{\bar{y}}_k(t+N_p|t)$ is calculated as

$$\hat{\bar{y}}_k(t+N_p|t) = C_k \prod_{i=t+N_p}^{t+1} A_k(i)x_k(t)$$
$$+ C_k \prod_{i=t+N_p}^{t+2} A_k(i)B_k(t+1)\bar{u}_k(t|t)$$
$$+ \cdots + C_k B_k(t+N_p)\bar{u}_k(t+N_p-1|t) \tag{14}$$

Therefore, the predictive output sequence of a batch process at time t is obtained as

$$\begin{aligned}\hat{\bar{Y}}_k(^{t+1}_{t+N_p}|t) &= \hat{\bar{Y}}_k(\bar{U}_k(^{t}_{t+N_p-1}|t)) \\ &= \Psi_k(t+1)x_k(t) + \hat{G}^N_k(t)\bar{U}_k(^{t}_{t+N_p-1}|t) \end{aligned} \quad (15)$$

where

$$\Psi_k(t) = [[C_kA_k(t)]^T, \cdots, [C_k\prod_{i=t+N_p-1}^{t}A_k(i)]^T]^T$$

$$\hat{G}^N_k(t) = \begin{bmatrix} \hat{G}_k(^{t+1}_N|t+1) \\ \hat{G}_k(^{t+1}_N|t+2) \\ \vdots \\ \hat{G}_k(^{t+1}_N|N) \end{bmatrix} = \begin{bmatrix} \hat{g}_k^{t+1,t+1} & 0 & \cdots & 0 \\ \hat{g}_k^{t+2,t+1} & \hat{g}_k^{t+2,t+2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \hat{g}_k^{N,t+1} & \hat{g}_k^{N,t+2} & \cdots & \hat{g}_k^{N,N} \end{bmatrix}$$

The tracking error of the prediction model for the remain trajectory is defined as

$$\begin{aligned}\hat{E}_k(^{t+1}_{t+N_p}|t) &= Y_d(t+1) - \hat{Y}_k(^{t+1}_{t+N_p}|t) \\ &= \bar{Y}_{k,d}(t+1) - \hat{\bar{Y}}_k(^{t+1}_{t+N_p}|t)\end{aligned} \quad (16)$$

where $Y_d(t+1) = [y_d(t+1), y_d(t+2), \cdots, y_d(t+N_p)]^T$, and $\bar{Y}_{k,d}(t+1) = [\bar{y}_{k,d}(t+1), \bar{y}_{k,d}(t+2), \cdots, \bar{y}_{k,d}(t+N_p)]^T$.

From Eqs. (12), (16), an iterative relationship between $\hat{E}_k(^{t+1}_{t+N_p}|t)$ and $E_{k-1}(t)$ can be derived as follows:

$$\begin{aligned}\hat{E}_k(^{t+1}_{t+N_p}|t) &= E_{k-1}(t+1) - \Psi_k(t+1)x_k(t) \\ &\quad - \hat{G}^N_k\Delta U_k(^{t}_{t+N_p-1}|t)\end{aligned} \quad (17)$$

where $E_{k-1}(t+1) = [e_{k-1}(t+1), \cdots, e_{k-1}(t+N_p)]^T$, and $\Delta U_k(^{t}_{t+N_p-1}|t) = [u_k(t|t) - u_{k-1}(t), \cdots, u_k(t+N_p-1|t) - u_{k-1}(t+N_p-1)]^T$.

Based on the tracking error of the process model, the moving quadratic criterion objective function can be constructed as follows:

$$\begin{aligned}&J_k(\Delta U_k(^{t}_{t+N_p-1}|t)) \\ &= \min_{\Delta U_k(^{t}_{t+N_p-1}|t)} \left\|\hat{E}_k(^{t+1}_{t+N_p}|t)\right\|^2_{Q_t} + \left\|\Delta U_k(^{t}_{t+N_p-1}|t)\right\|^2_{R_t}\end{aligned} \quad (18)$$

Q_t and R_t are both weighting matrices defined as $Q_t = q \times I_{N_p\times N_p}$ and $R_t = r \times I_{N_p\times N_p}$, where q and r are both positive real numbers.

In industry control applications, the following constraint of control input is usually considered.

$$U^{\text{low}} \le U_k \le U^{\text{up}} \quad (19)$$

The inequality (19) can be rewritten as

$$\Phi\Delta U_k \ge \Omega_k \quad (20)$$

where

$$\Phi = [I \ -I]^T$$
$$\Omega_k = \begin{bmatrix} U^{\text{low}} - U_{k-1} \\ -(U^{\text{up}} - U_{k-1}) \end{bmatrix}$$

U^{low} and U^{up} are the lower and upper bounds of the input sequence, respectively.

The optimization problem (18) can be cast into the following formulation.

$$\begin{aligned}&\min_{\lambda,\Delta U_k(^{t}_{t+N_p-1}|t)} \lambda \\ &\text{s.t.} \ \left\|\hat{E}_k(^{t+1}_{t+N_p}|t)\right\|^2_{Q_t} + \left\|\Delta U_k(^{t}_{t+N_p-1}|t)\right\|^2_{R_t} \le \lambda\end{aligned} \quad (21)$$

Using Schur complement [28], the constraint (21) can be converted to an LMI as follows:

$$\begin{bmatrix} \lambda & \hat{E}_k(^{t+1}_{t+N_p}|t)^T & \Delta U_k(^{t}_{t+N_p-1}|t)^T \\ * & Q_t^{-1} & 0 \\ * & * & R_t^{-1} \end{bmatrix} \ge 0 \quad (22)$$

Thus, the control input with constraint can be obtained by solving the following convex optimization problem.

$$\begin{aligned}&\min_{\lambda,\Delta U_k(^{t}_{t+N_p-1}|t)} \lambda \\ &\text{s.t.} \begin{bmatrix} \lambda & \hat{E}_k(^{t+1}_{t+N_p}|t)^T & \Delta U_k(^{t}_{t+N_p-1}|t)^T \\ * & Q_t^{-1} & 0 \\ * & * & R_t^{-1} \end{bmatrix} \ge 0\end{aligned} \quad (23)$$

$$\begin{bmatrix} I_{N_p\times N_p} \\ -I_{N_p\times N_p} \end{bmatrix}\Delta U_k(^{t}_{t+N_p-1}|t) \ge \begin{bmatrix} U^{\text{low}}(t) - U_{k-1}(t) \\ -U^{\text{up}}(t) + U_{k-1}(t) \end{bmatrix} \quad (24)$$

Inequalities (23) and (24) are the LMIs in regard to $\Delta U_k(^{t}_{t+N_p-1}|t)$. The control law is directly updated based on the information obtained from the current and the pervious batch runs. In summary, the proposed QMPILC algorithm is given as follows:

Step 1: Set $k = 1$. Calculate the control action for the first batch based on the local inverse model of the system, and initialize Q_t and R_t.

Step 2: After the completion of the kth batch run, we can obtain the input sequence U_k and the output sequence Y_k. Then set U_k and Y_k as the nominal input and the nominal output of the next batch, respectively. According to the historical process operation data, the deviation model of the next batch is identified by using the JITL method.

Step 3: Set $t = 1$. At time t in the $(k+1)$th batch, the predictive control sequence $U_{k+1}(^{t}_{t+N_p-1}|t)$ can be obtained by solving LMIs (23) and (24). Then the first element $u_{k+1}(t|t)$ of the sequence $U_{k+1}(^{t}_{t+N_p-1}|t)$ is implemented to the control system.

Step 4: If $t < N$, set $t = t+1$ and go back to Step 3, else set $k = k+1$ and go to Step 2.

IV. THE ANALYSIS OF CONVERGENCE

In this section, we will give the analyses of the convergence and stability based on the proposed control algorithm.

(1) Perfect model

Theorem 1: Consider a batch process described by Eq. (1) controlled by the proposed control algorithm. Under assumptions A1 − A2, the control sequence will converge to a constant, and the tracking error sequence E_k will converge to zero if there is no modeling error. That is, $\Delta U_k = U_k - U_{k-1} \rightarrow 0$ and $E_k \rightarrow 0$ as $k \rightarrow \infty$.

Proof: Let

$$U_k(^{t1}_{t2}) = [u_k(t_1), u_k(t_1+1), \cdots, u_k(t_2)]^T$$

$$\Delta U_k(^{t1}_{t2}) = U_k(^{t1}_{t2}) - U_{k-1}(^{t1}_{t2})$$

$$V_k = \left\|\hat{E}_k\right\|^2_{Q_0} + \|\Delta U_k\|^2_{R_0}$$

$$\hat{G}^P_k(t) = [\hat{G}_k(^1_t|t+1)^T, \hat{G}^T_k(^1_t|t+2)^T, \cdots \hat{G}^T_k(^1_t|N)^T]^T$$

where $\hat{G}_k(^1_t|t+i) = [\hat{g}^{t+i,1}_k, \hat{g}^{t+i,2}_k, \cdots, \hat{g}^{t+i,t}_k]$.

According to Eqs. (9) and (12), Eq. (18) can be rewritten as

$$\begin{aligned}
&J_k(\Delta U_k(^t_{t+N_p-1}|t))\\
&= \left\|E_{k-1}(t+1) - \hat{G}^P_k(t)\Delta U_k(^0_{t-1}) - \hat{G}^N_k(t)\Delta U_k(^t_{N-1}|t)\right\|^2_{Q_t}\\
&\quad + \left\|\Delta U_k(^t_{N-1}|t)\right\|^2_{R_t}\\
&= \left\|e_{k-1}(t+1) - \hat{G}_k(^1_t|t+1)\Delta U_k(^0_{t-1}) - \hat{g}^{t+1,t+1}_k\Delta u(t|t)\right\|^2_q\\
&\quad + \|\Delta u_k(t|t)\|^2_r + \sum_{i=2}^{N-t}\left\|e_{k-1}(t+i) - \hat{G}_k(^1_t|t+i)\Delta U_k(^0_{t-1})\right.\\
&\quad \left. - \hat{G}_k(^{t+1}_N|t+i)\Delta U_k(^t_{N-1}|t)\right\|^2_q + \|\Delta u_k(t+i-1|t)\|^2_r
\end{aligned} \tag{25}$$

At time $t+1$, $\Delta U_k(^{t+1}_{N-1}|t+1)$ is the optimal solution of the objective function (18), and due to $u(t) = u(t|t)$, the following inequality holds.

$$\begin{aligned}
&J_k(\Delta U_k(^{t+1}_{N-1}|t+1))\\
&= \left\|E_{k-1}(t+2) - \hat{G}^P_k(t+1)\Delta U_k(^0_t)\right.\\
&\quad \left. - \hat{G}^N_k(t+1)\Delta U_k(^{t+1}_{N-1}|t+1)\right\|^2_{Q_{t+1}} + \left\|\Delta U_k(^{t+1}_{N-1}|t+1)\right\|^2_{R_{t+1}}\\
&\le J_k(\Delta U_k(^{t+1}_{N-1}|t)) = \left\|E_{k-1}(t+2) - \hat{G}^P_k(t+1)\Delta U_k(^0_t)\right.\\
&\quad \left. - \hat{G}^N_k(t+1)\Delta U_k(^{t+1}_{N-1}|t)\right\|^2_{Q_{t+1}} + \left\|\Delta U_k(^{t+1}_{N-1}|t)\right\|^2_{R_{t+1}}\\
&= \sum_{i=2}^{N-t}\left\|e_{k-1}(t+i) - \hat{G}_k(^1_t|t+i)\Delta U_k(^0_{t-1})\right.\\
&\quad \left. - \hat{G}_k(^{t+1}_N|t+i)\Delta U_k(^t_{N-1}|t)\right\|^2_q + \|\Delta u_k(t+i-1|t)\|^2_r
\end{aligned} \tag{26}$$

From Eqs. (25) and (26), we get

$$\begin{aligned}
&\left\|e_{k-1}(t+1) - \hat{G}_k(^1_t|t+1)(t)\Delta U_k(^0_{t-1}) - \hat{g}^{t+1,t+1}_k\Delta u_k(t|t)\right\|^2_q\\
&\quad + \|\Delta u_k(t|t)\|^2_r + \left\|E_{k-1}(t+2) - \hat{G}^P_k(t+1)\Delta U_k(^0_t)\right.\\
&\quad \left. - \hat{G}^N_k(t+1)\Delta U_k(^{t+1}_{N-1}|t+1)\right\|^2_{Q_{t+1}} + \left\|\Delta U_k(^{t+1}_{N-1}|t+1)\right\|^2_{R_{t+1}}\\
&\le \left\|E_{k-1}(t+1) - \hat{G}^P_k(t)\Delta U_k(^0_{t-1}) - \hat{G}^N_k(t)\Delta U_k(^t_{N-1}|t)\right\|^2_{Q_t}\\
&\quad + \left\|\Delta U_k(^t_{N-1}|t)\right\|^2_{R_t}
\end{aligned} \tag{27}$$

According to Eq. (27), after N iterations, the following inequality is obtained.

$$\begin{aligned}
V_k &\le J_k(\Delta U_k(^0_{N-1}|t))\\
&= \left\|E_{k-1} - \hat{G}_k\Delta U_k(^0_{N-1}|0)\right\|^2_{Q_0} + \left\|\Delta U_k(^0_{N-1}|0)\right\|^2_{R_0}
\end{aligned} \tag{28}$$

Since $\Delta U_k(^0_{N-1}|0)$ is the optimal solution of the objective function (18) at time $t = 0$, we have

$$\begin{aligned}
&\left\|E_{k-1} - \hat{G}_k\Delta U_k(^0_{N-1}|0)\right\|^2_{Q_0} + \left\|\Delta U_k(^0_{N-1}|0)\right\|^2_{R_0}\\
&\quad \le J_k(0) = \|E_{k-1}\|^2_{Q_0} = V_{k-1} - \|\Delta U_{k-1}\|^2_{R_0}
\end{aligned} \tag{29}$$

From inequality (29), we have

$$V_k \le V_{k-1} - \|\Delta U_{k-1}\|^2_{R_0} \tag{30}$$

Inequality (30) leads to

$$V_k + \sum_{j=1}^{k-1}\Delta U_j^T R_0 \Delta U_j \le V_1 < \infty \tag{31}$$

Since $\Delta U_j^T R_0 \Delta U_j \ge 0$ and the sequence $\{\sum_{j=1}^{k-1}\Delta U_j^T R_0 \Delta U_j\}$ is non-decreasing, we can conclude that the sequence $\{\sum_{j=1}^{k-1}\Delta U_j^T R_0 \Delta U_j\}$ converges and the following equation holds.

$$\begin{aligned}
&\lim_{k\to\infty}\Delta U_{k-1}^T R_0 \Delta U_{k-1}\\
&\quad = \lim_{k\to\infty}\left(\sum_{j=1}^{k-1}\Delta U_j^T R_0 \Delta U_j - \sum_{j=1}^{k-2}\Delta U_j^T R_0 \Delta U_j\right)\\
&\quad = 0
\end{aligned} \tag{32}$$

It implies that

$$\lim_{k\to\infty}\Delta U_k = 0 \tag{33}$$

Thus, we have $E_k \rightarrow E_\infty$ as $k \rightarrow \infty$. Because $\Delta U_k(^t_{N-1}|t)$ is the optimal solution of the objective function at time t, it leads to

$$\begin{aligned}
&\frac{1}{2}\frac{\partial J_k(\Delta U_k(^t_{N-1}|t))}{\partial \Delta U_k(^t_{N-1}|t)}\\
&\quad = -\hat{G}^N_k(t)^T Q_t(E_{k-1}(t+1) - \hat{G}^P_k(t)\Delta U_k(^0_{t-1}))\\
&\quad\quad + (\hat{G}^N_k(t)^T Q_t \hat{G}^N_k(t) + R_t)\Delta U_k(^t_{N-1}|t) = 0
\end{aligned} \tag{34}$$

According to Eq. (12) and $E_k = \hat{E}_k$, we get

$$\lim_{k\to\infty}(E_{k-1} - E_k) = \lim_{k\to\infty} \hat{G}_k \Delta U_k \tag{35}$$

From Eqs. (33), (35) and (34), we have

$$\begin{aligned} &-\hat{G}^N_\infty(t)^T Q_t E_\infty(t+1) \\ &\quad + (\hat{G}^N_\infty(t)^T Q_t \hat{G}^N_\infty(t) + R_t)\Delta U_\infty({}^t_{N-1}|t) = 0 \end{aligned} \tag{36}$$

Because $\hat{G}_k$ is a full row-rank and R_t and Q_t are positive definite matrices in the objective function, Eq. (33) and Eq. (36) result in

$$\lim_{k\to\infty} E_k = 0 \tag{37}$$

Therefore, if there is no modeling error, zero tracking error will be achieved under the proposed control method.

(2) Model-plant mismatch

Theorem 2: Considering that there exist the model errors between the prediction model and the actual process. Under assumptions A1 − A2, the tracking error E_k of the proposed optimization strategy can be bounded in a small region, namely, $E_k \to \epsilon$ as $k \to \infty$, where ϵ is a small positive constant.

Proof: The prediction error of the output is defined as

$$\varepsilon_k = \hat{Y}_k - Y_k \tag{38}$$

In this study, since the prediction error is bound by a small positive constant, we have

$$|\varepsilon_k| < \delta \tag{39}$$

According to Eq. (36), we get

$$-\hat{G}_k^T Q_0 E_{k-1} + (\hat{G}_k^T Q_0 \hat{G}_k + R_0)\Delta U_k({}^0_{N-1}|0) = 0 \tag{40}$$

According to Eqs. (12), (18) and (38), the update model for tracking error based on the actual process can be obtained as

$$E_k = E_{k-1} - \hat{G}_k \Delta U_k + \varepsilon_k \tag{41}$$

From Eqs. (33), (40) and (41), it follows that

$$\lim_{k\to\infty} E_k = \lim_{k\to\infty} \varepsilon_k \tag{42}$$

Because ε_k is bound, the tracking error E_k can converge to a small region, that is, $E_k \to \epsilon$ as $k \to \infty$. Therefore, the tracking performance of the system depends on the accuracy of the prediction model.

V. PERFORMANCE ANALYSIS AND DISCUSSION

In section, two cases will be conducted to show the effectiveness of the proposed methods. It assumes that the mechanistic model of process is unavailable and the disturbance is unmeasurable.

A. CASE 1

Consider the batch process described by the following linear time-varying model. The terminal time is $t_f = 50$ and the sampling period is $t_s = 1$.

$$y_k(t) = \frac{K(t)}{T(t)s + 1} u_k(t) \tag{43}$$

where

$$\begin{aligned} T(t) &= 0.001t^2 + 3 \\ K(t) &= -0.03t^2 + 1.7t + 5 \end{aligned}$$

The desired reference trajectory is

$$y_d(t) = \begin{cases} 0.4t & t \in [0, 15] \\ 6 & t \in [15, 25] \\ 6 - 0.2(t - 25) & t \in [25, 40] \\ 3 & t \in [40, 50] \end{cases} \tag{44}$$

The discretization of the process model is described as follows:

$$y_k(t) = \frac{b_1(t)q^{-1}}{1 + a_1(t)q^{-1}} u_k(t) \tag{45}$$

where

$$a_1(t) = -\exp(-t_s/T(t)) \tag{46}$$
$$b_1(t) = K(t)(1 - \exp(-t_s/T(t))) \tag{47}$$

The design parameters are given as follows: $q = 1$ and $r = 10$. The root mean square error (RMSE) of the tracking error E_k is used to show the tracking performance. Figs. 2 and 3 depict output and input of the control system, respectively. Fig. 4 shows the RMSE of the tracking error. In order to test the ability of rejecting disturbance, the disturbance whose value is 0.6 is added into the output at time $t = 20$ in the 7th batch. The output and input are shown in Figs. 5 and 6, respectively. The tracking performance of the system with external disturbance is shown in Fig. 7.

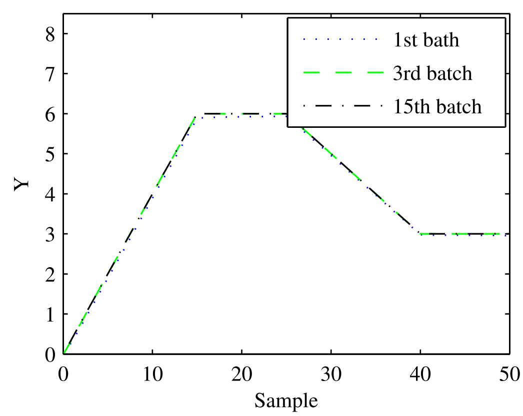


FIGURE 2. Output trajectories in the 1st, the 3rd and the 15th batches of the system.

From Figs. 2 and 3, a fast convergence rate is obtained by using the proposed control strategy. Figs. 5 and 7 show that the proposed control system provides the ability of rejecting disturbance, and output still converge to the reference trajectory after the disturbance added into the system.

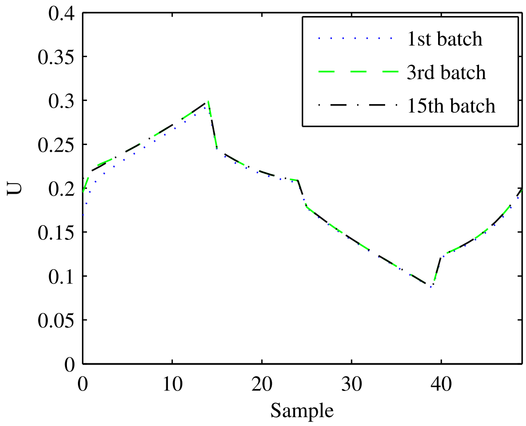


FIGURE 3. Input trajectories in the 1st, the 3rd and the 15th batches of the system.

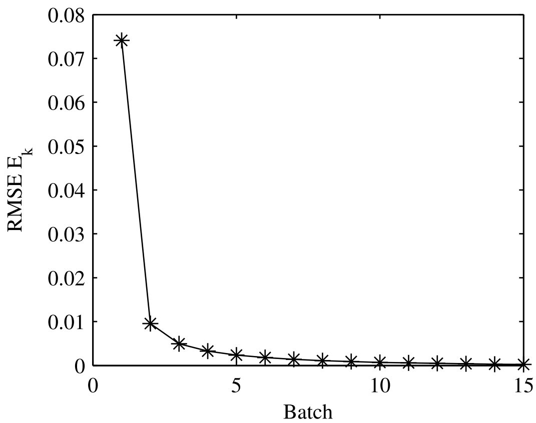


FIGURE 4. Curve of RMSE of E_k.

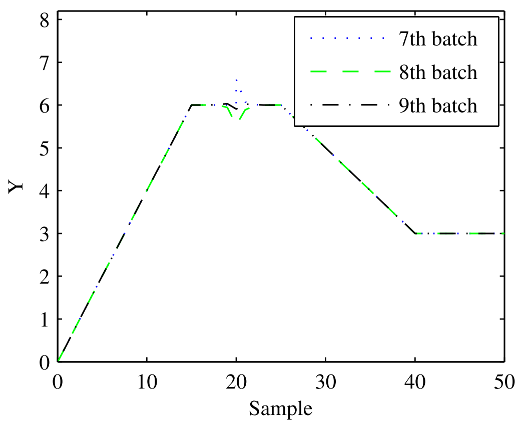


FIGURE 5. Output trajectories in the 7th, the 8th and the 9th batches of the system with external disturbance.

B. CASE 2

The proposed control strategy is applied to control a typical nonlinear batch rector, in which a first-order irreversible exothermic reaction $A \rightarrow B \rightarrow C$ take place [30]. The following dynamic equations describe the

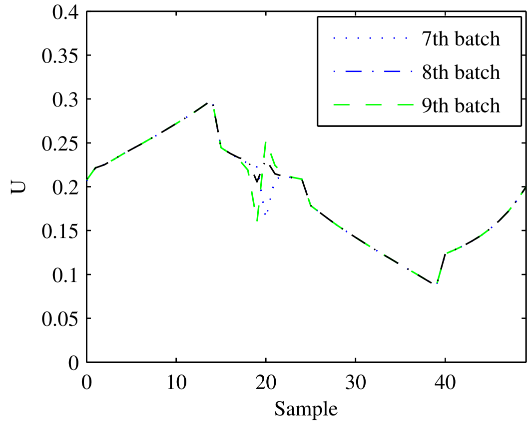


FIGURE 6. Input trajectories in the 7th, the 8th and the 9th batches of the system with external disturbance.

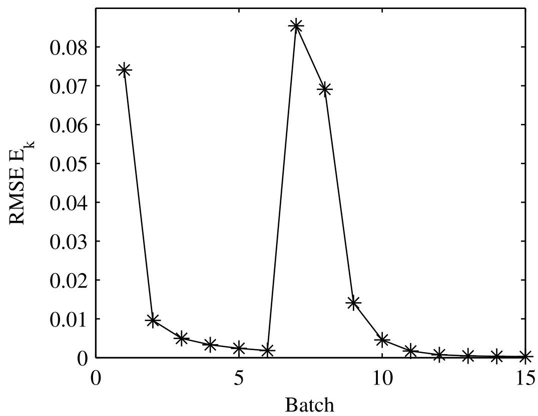


FIGURE 7. Curve of RMSE of E_k for the system with external disturbance.

reaction process.

$$\dot{x}_1 = -k_1 \exp(-E_1/T)x_1^2 \tag{48}$$

$$\dot{x}_2 = k_1 \exp(-E_1/T)x_1^2 - k_2 \exp(-E_2/T)x_2 \tag{49}$$

where x_1 and x_2 represent the reactant concentration of A and B, respectively, and T denote the reaction temperature. The final time t_f is fixed to be 1.0 h. The values of parameters k_1, k_2, E_1 and E_2 are given in Table 1.

TABLE 1. Parameter values for the batch reactor.

parameters	values
k_1	4.0×10^3
k_2	6.2×10^5
E_1	2.5×10^3
E_2	5×10^3

In this reaction, the reactor temperature is divided into 10 equal intervals, and it is normalized by using $u = (T - T_{\min})/(T_{\max} - T_{\min})$, in which $T_{\min}$ and $T_{\max}$ are 298(K) and 398(K), respectively. u is the control variable which is bounded as $0 \leq u \leq 1$, and $x_2(t)$ is the output variable. The control objective is to minimize the end-time tracking error by adjusting the control input.

The initial operating conditions are $x_1(0) = 1$ and $x_2(0) = 0$. The ideal value of end-time output is $y_d(t_f) = 0.61$. The parameters of the control system are chosen as follows: $q = 1$ and $r = 0.0001$.

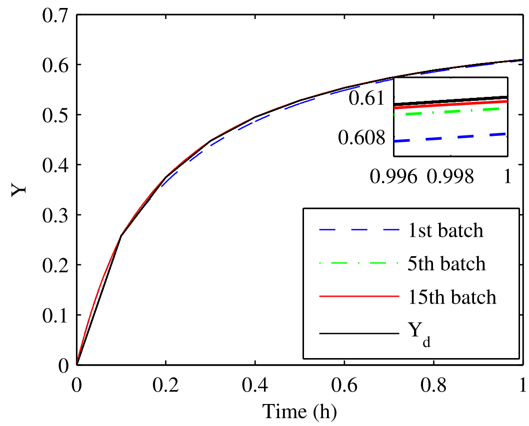


FIGURE 8. Trajectories of product quality variable in the 1st, the 5th and the 15th batches.

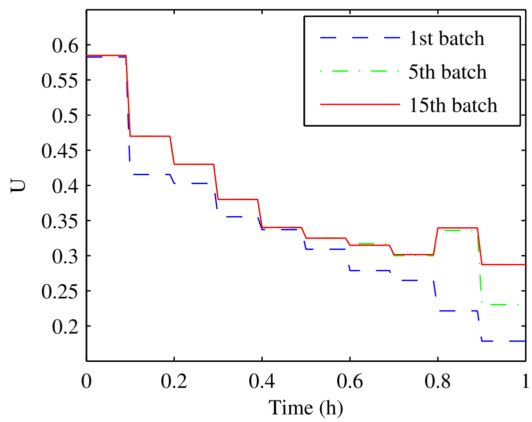


FIGURE 9. Input temperature profiles in the 1st, the 5th and the 15th batches.

The proposed control strategy is compared with integrated MPC strategy [22] and batch to batch ILC [29]. The output trajectories in the 1st, the 5th and the 15th batches and the corresponding input trajectories of the system are shown in Figs. 8 and 9, respectively. The tracking performance of the proposed control strategy is shown in Fig. 10. In Fig. 10, output trajectories converge to the reference trajectory after two batches. Fig. 11 shows the endpoint tracking error values based on three control strategies. Since the input of the first batch is obtained by using the inverse model of the system, the proposed strategy provides a faster convergence rate than integrated MPC and batch to batch ILC. The majority of the process nonlinearity is removed by subtracting the nominal trajectories from the batch operation trajectories in our scheme, which result in a higher precision of the model in this paper than that in [22]. Thus convergence accuracy is improved. Moreover, the control input can be obtained

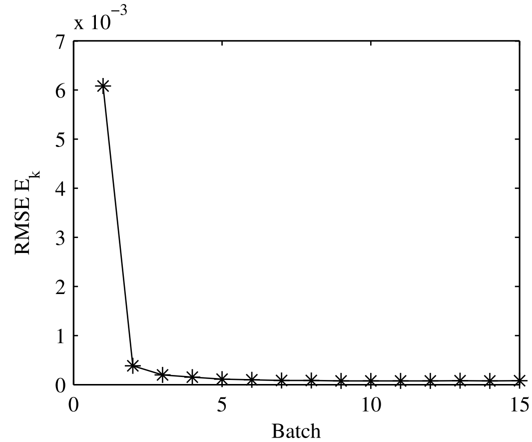


FIGURE 10. Tracking performance of the system based on the proposed control strategy.

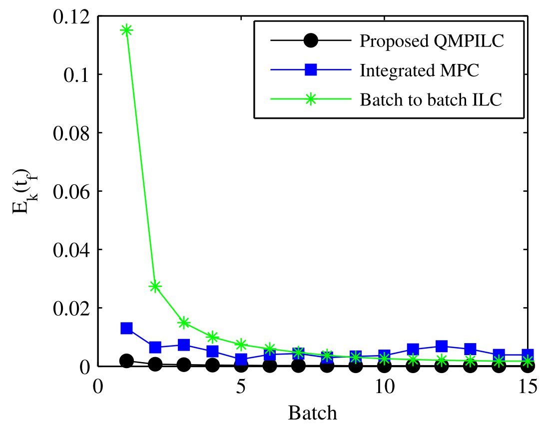


FIGURE 11. Comparisons of endpoint errors based on three control strategies.

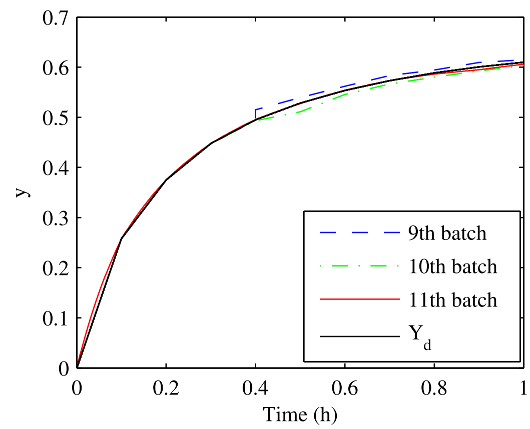


FIGURE 12. Trajectories of product quality variable in the 9th, the 10th and the 11th batches of the system with external disturbance.

by solving a convex optimization problem, which leads to a higher operational efficiency than that of integrated MPC strategy.

The external disturbance with value being 0.02 at time $t = 0.4$ h in the 9th batch is added to the output of

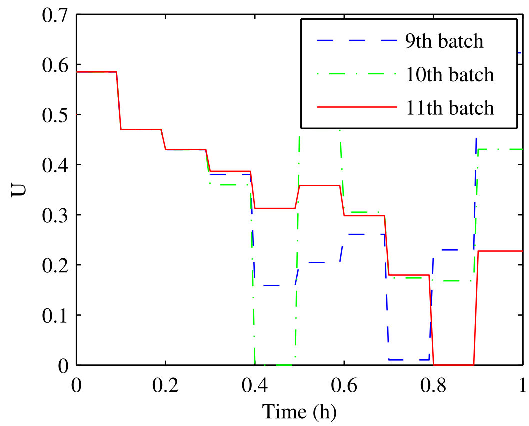


FIGURE 13. **Input temperature profiles in the 9th, the 10th and the 11th batches of the system with external disturbance.**

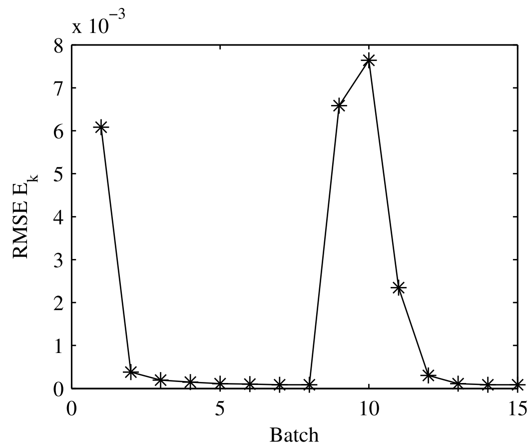


FIGURE 14. **Tracking performance of the system with external disturbance.**

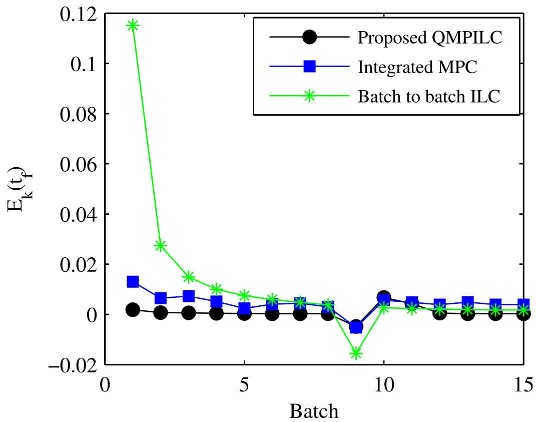


FIGURE 15. **Comparison of endpoint errors based on three control strategies with external disturbance.**

the system. In Figs. 12 and 14, due to a real-time feedback loop, the control system can reject disturbance, and the output still converges to the reference trajectory in the batch direction. Fig. 15 shows the ability of rejecting the disturbance based on three control methods. It indicates the proposed control strategy provides a better ability on disturbance rejection compared with the methods in [22] and [29].

VI. CONCLUSION

A model predictive iterative learning control algorithm based on quadratic-criterion for batch processes has been proposed in this paper. By using the JITL method, a model whose parameters vary with time and batch can well describe nonlinear batch process. A quadratic-criterion-based model predictive iterative learning control design has been given, and the control law has been obtained by solving a convex optimization problem. The analysis of convergence has been also provided. The proposed methods have been illustrated on batch processes, and the results show that system performance has been improved.

REFERENCES

[1] D. Shen and Y. Wang, "Survey on stochastic iterative learning control," *J. Process Control*, vol. 24, no. 12, pp. 64–77, Dec. 2014.

[2] Y. Wang, H. Zhang, S. Wei, D. Zhou, and B. Huang, "Control performance assessment for ILC-controlled batch processes in a 2-D system framework," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 48, no. 9, pp. 1493–1504, Sep. 2018.

[3] D. Shen, J. Han, and Y. Wang, "Stochastic point-to-point iterative learning tracking without prior information on system matrices," *IEEE Trans. Autom. Sci. Eng.*, vol. 14, no. 1, pp. 376–382, Jan. 2017.

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

[5] Y. Wang, F. Gao, and F. J. 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.

[6] Z. Ge, "Process data analytics via probabilistic latent variable models: A tutorial review," *Ind. Eng. Chem. Res.*, vol. 57, pp. 12646–12661, Aug. 2018.

[7] J. H. Lee and K. S. Lee, "Iterative learning control applied to batch processes: An overview," *Control Eng. Pract.*, vol. 15, no. 10, pp. 1306–1318, 2007.

[8] R. Chi, X. Liu, R. Zhang, Z. Hou, and B. Huang, "Constrained data-driven optimal iterative learning control," *J. Process Control*, vol. 55, pp. 10–29, Jul. 2017.

[9] D. H. Nguyen and D. Banjerdpongchai, "A convex optimization approach to robust iterative learning control for linear systems with time-varying parametric uncertainties," *Automatica*, vol. 47, no. 9, pp. 2039–2043, Sep. 2011.

[10] Z. Xiong, J. Zhang, and J. Dong, "Optimal iterative learning control for batch processes based on linear time-varying perturbation model," *Chin. J. Chem. Eng.*, vol. 16, no. 2, pp. 235–240, 2008.

[11] Z. Xu, J. Zhao, Y. Yang, Z. Shao, and F. Gao, "Optimal iterative learning control based on a time-parametrized linear time-varying model for batch processes," *Ind. Eng. Chem. Res.*, vol. 52, no. 18, pp. 6182–6192, Apr. 2013.

[12] D. Shen, J. Han, and Y. Wang, "Convergence analysis of ILC input sequence for underdetermined linear systems," *Sci. China Inf. Sci.*, vol. 60, no. 9, pp. 305–307, Sep. 2017.

[13] D. Shen, W. Zhang, Y. Wang, and C.-J. Chien, "On almost sure and mean square convergence of P-type ILC under randomly varying iteration lengths," *Automatica*, vol. 63, pp. 359–365, Jan. 2016.

[14] S. Mo, L. Wang, Y. Yao, and F. Gao, "Two-time dimensional dynamic matrix control for batch processes with convergence analysis against the 2D interval uncertainty," *J. Process Control*, vol. 22, pp. 899–914, Jun. 2012.

[15] T. Liu, X. Z. 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.

[16] Y. Wang, D. Zhou, and F. Gao, "Iterative learning model predictive control for multi-phase batch processes," *J. Process Control*, vol. 18, pp. 543–557, Jul. 2008.

[17] S.-K. Oh and J. M. Lee, "Iterative learning model predictive control for constrained multivariable control of batch processes," *Comput. Chem. Eng.*, vol. 93, no. 4, pp. 284–292, Oct. 2016.

[18] J. Lu, Z. Cao, Z. Wang, and F. Gao, "A two-stage design of two-dimensional model predictive iterative learning control for nonrepetitive disturbance attenuation," *Ind. Eng. Chem. Res.*, vol. 54, pp. 5683–5689, May 2015.

[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] Y. Wang, T. Liu, and Z. Zhao, "Advanced PI control with simple learning set-point design: Application on batch processes and robust stability analysis," *Chem. Eng. Sci.*, vol. 71, no. 26, pp. 153–165, Mar. 2012.

[21] K. S. Lee, I.-S. Chin, H. J. Lee, and J. H. Lee, "Model predictive control technique combined with iterative learning for batch processes," *AIChE J.*, vol. 45, no. 10, pp. 2175–2187, Oct. 1999.

[22] L. Jia and W. Tan, "Just-in-time learning based integrated MPC-ILC control for batch processes," *Chin. J. Chem. Eng.*, vol. 26, no. 8, pp. 1713–1720, Aug. 2018.

[23] C. Han, L. Jia, and D. Peng, "Model predictive control of batch processes based on two-dimensional integration frame," *Nonlinear Anal., Hybrid Syst.*, vol. 28, pp. 75–86, May 2018.

[24] Q. Su, M.-S. Chiu, and R. D. Braatz, "Integrated B2B-NMPC control strategy for batch/semibatch crystallization processes," *AIChE J.*, vol. 63, no. 11, pp. 5007–5018, Nov. 2017.

[25] C. Cheng and M.-S. Chiu, "Robust PID controller design for nonlinear processes using JITL technique," *Chem. Eng. Sci.*, vol. 63, no. 21, pp. 5141–5148, Nov. 2008.

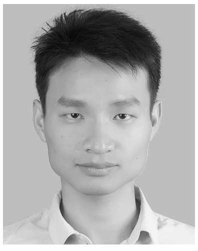
[26] C. Cheng and M.-S. Chiu, "A new data-based methodology for nonlinear process modeling," *Chem. Eng. Sci.*, vol. 59, no. 13, pp. 2801–2810, Jul. 2004.

[27] M. Arif, T. Ishihara, and H. Inooka, "Incorporation of experience in iterative learning controllers using locally weighted learning," *Automatica*, vol. 37, no. 6, pp. 881–888, Jun. 2001.

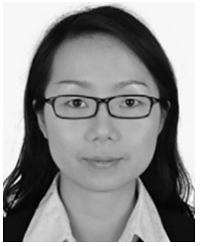
[28] S. Boyd, L. E. Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*. Philadelphia, PA, USA: SIAM, 1994.

[29] L. Jia, J. P. Shi, and M.-S. Chiu, "Integrated neuro-fuzzy model and dynamic R-parameter based quadratic criterion-iterative learning control for batch process," *Neurocomputing*, vol. 98, no. 3, pp. 24–33, Dec. 2012.

[30] J. S. Logsdon and L. T. Biegler, "Accurate solution of differential-algebraic optimization problems," *Ind. Eng. Chem. Res.*, vol. 28, no. 11, pp. 1628–1639, Nov. 1989.



LIUMING ZHOU received the B.S. degree in automation from Huainan Normal University, in 2012, and the M.S. degree in control theory and control engineering from the Jiangsu University of Science and Technology, in 2015. He is currently pursuing the Ph.D. degree with Shanghai University. His research interests include iterative learning control for batch processes and robust control.



LI JIA received the Ph.D. degree in control theory and control engineering from the East China University of Science and Technology, China, in 2003. She was a Research Fellow with the National University of Singapore, from March 2003 to June 2005. She is currently a Professor with the School of Mechatronics Engineering and Automation, Shanghai University. Her research interests include the areas of fuzzy control, neural networks, and intelligent tools, which are toward the development of innovative identification and control strategies for complex process systems.

YU-LONG WANG (M'14) received the B.S. degree in computer science and technology from Liaocheng University, Liaocheng, China, in 2000, and the M.S. and Ph.D. degrees in control science and engineering from Northeastern University, Shenyang, China, in 2006 and 2008, respectively.

He was a Postdoctoral Research Fellow and a Research Fellow with Central Queensland University, North Rockhampton, QLD, Australia, an Academic Visitor with The University of Adelaide, Adelaide, SA, Australia, and a Professor with the Jiangsu University of Science and Technology, Zhenjiang, Jiangsu, China. In 2017, he was appointed as an Eastern Scholar by the Municipal Commission of Education, Shanghai, China, and joined Shanghai University, Shanghai, China, where he is currently a Professor. His current research interests include networked control systems, fault detection, and the motion control for marine vehicles.