2025 International Conference on Intelligent Computing
July 26-29, Ningbo, China
https://www.ic-icc.cn/2025/index.php

Q-learning-based Optimal Control Scheme for Time-varying Uncertain Batch Processes

Jianan Liu[1,2,3], Wenjing Hong[1,2,3], Jia Shi[1,2,3]✉

[1] Institute of Artificial Intelligence, Xiamen University, Xiamen 316000, China
[2] Department of Chemical and Biochemical Engineering, Xiamen University, Xiamen 316000, China
[3]Innovation Laboratory for Sciences and Technologies of Energy Materials of Fujian Province (IKKEM), Xiamen University, Xiamen 316000, China
jshi@xmu.edu.cn

Abstract. Industrial batch processes are extensively employed in the modern manufacturing industry due to their efficiency and flexibility. However, controlling and optimizing batch processes is difficult because of their complex non-stationary dynamics and inherent time-varying uncertainties. In order to address these issues, this paper proposes a Q-learning-based optimal control scheme for time-varying batch processes to achieve optimal control while reducing reliance on process modeling. Based on the time-varying nominal model, an initial control policy is derived from the principle of optimality and dynamic programming. Nevertheless, the presence of unknown time-varying system uncertainties hinders the optimal performance of the initial control policy. To overcome this limitation, we utilize the repetitive nature of the batch process to collect operational data from multiple batches runs under the initial control policy. Then, the Q-learning-based optimal control scheme is developed to iteratively improve the initial control policy under the reinforcement learning framework. Finally, the results from the experimental simulations in the numerical multi-input multi-output batch system and the injection molding process confirm the efficiency, applicability, and superior control performance.

Keywords: Reinforcement Q-learning, Optimal Control Scheme, Time-varying batch processes, Time-varying system uncertainties.

1 Introduction

Batch processes are repetitive operations that execute the same tasks over a finite period. Due to their notable advantages, such as low capital costs, less material, and flexible production conversion, batch processes such as robotic manipulators [1,2], chemical industries [3,4], and integrated circuits [5] are extensively employed in modern manufacturing. However, controlling batch processes is extremely challenging due to the increasing complexity of production requirements, particularly in complex system dynamics, model uncertainties, and changing operating conditions. Over the past decades, many advanced control strategies have been developed for batch processes, including iterative learning control (ILC) [6,7], model predictive control (MPC) [8,9], and reinforcement learning control (RLC) [10]. Nevertheless, most existing works have concentrated on traditional time-invariant batch processes that are impacted by external perturbations, model errors, and environmental noise [11,12]. In contrast, limited attention has been given to practical time-varying batch processes, such as injection molding [13] and continuous stirred tank reactors [14]. Time-invariant control strategies often fail to ensure adequate performance and robustness in time-varying batch processes.

For time-varying batch processes, Owens proposed a least-norm ILC to achieve high-precision trajectory tracking [15], while Sebastian enhanced the D-type ILC by incorporating a feedback loop to improve control performance [16]. Additionally, input-constrained MPC has been developed to ensure asymptotic stability in multi-input multi-output (MIMO) linear time-varying (LTV) batch processes [17]. Although these control schemes demonstrate effective control performance, they rely on the accuracy of the system model. In practical applications, there are inevitable modeling errors between the actual system and the nominal model. These modeling errors can significantly degrade control performance and compromise system stability and safety.

Numerous control schemes combined with robust control theory have been proposed to address the inevitability of the model uncertainties. The fundamental idea of robust control is to minimize the maximum possible control error in the presence of bounded modeling uncertainties. Jan proposed a robust ILC to ensure monotonic convergence for the time-varying batch processes [18]. Hao developed a novel PI-type indirect ILC by integrating PI-type ILC with robust control to address the time-varying model uncertainties [19]. Additionally, Zhang extended MPC by employing a systematic min-max optimization method to counteract system uncertainties [20]. The above robust control schemes can effectively suppress the system uncertainties and guarantee control performance. However, robust control approaches assume known limitations for the unknown uncertainties. A significant uncertainty boundary makes the robust controller conservative, preventing optimal control performance, whereas a small uncertainty boundary hinders robustness and can not effectively eliminate system uncertainties. Thus, designing the optimal control scheme for time-varying batch processes with time-varying uncertainties remains a significant challenge.

Reinforcement learning (RL) has emerged as a powerful method for optimizing intricate dynamic systems. Unlike model-based control schemes, RL directly interacts with the controlled systems to learn unknown system uncertainties and achieve the optimal control objective. Without system dynamics, Lopez proposed an efficient off-policy Q-learning method for infinite discrete time-invariant systems to achieve optimal control performance [21]. A two-dimensional ILC scheme based on the Q-learning algorithm was proposed for linear time-invariant batch processes to perform high-precision control [22]. These control schemes highlight reinforcement learning as an effective advanced technology that can enhance traditional model-based control methods and restrain the negative influence of system uncertainties in time-varying batch processes.

Inspired by the principle of RL and the repetitive nature of batch processes, this paper proposes a novel Q-learning-based optimal control scheme for time-varying batch processes with time-varying system uncertainties. Initially, a non-optimal control policy is derived from the principle of optimality and dynamic programming based on an inaccurate nominal model. Using the repetitive nature and the off-policy Q-learning algorithm, we develop a Q-learning-based optimal control scheme to improve the initial non-optimal control policy. Finally, the effectiveness and the control performance of the proposed control method are validated in the linear MIMO time-varying batch system and the time-varying injection molding process. The key innovations and contributions of this work are as follows:

1. We propose a novel data-driven optimal control scheme for time-varying batch processes with time-varying uncertainties by integrating the Q-learning algorithm with dynamical programming.
2. Unlike the previous time-varying batch control schemes, which depend on the model accuracy [15,16,17], or the robust control schemes [18,19,20], which rely on the prior knowledge of the uncertainty boundary, the proposed model-free control scheme only utilizes the batch data to achieve optimal control policy.
3. Compared with the reinforcement learning control schemes for time-invariant batch processes [21,22], the Q-learning-based optimal control scheme provides an effective control scheme for time-varying batch processes.

The structure of this paper is outlined as follows: **Section 2** includes the problem description along with the design objectives for the optimal control scheme. **Section 3** provides a comprehensive explanation of the proposed Q-learning-based optimal control scheme. In **Section 4**, simulations of the proposed control scheme are conducted on the MIMO time-varying batch system and the injection molding process to demonstrate outstanding control performance. The paper finally concludes with important conclusions and insights in **Section 5**.

2 Problem Description

It is assumed that the industrial batch process with time-varying system dynamics and system uncertainties can be described by:

$$\begin{cases} x_{k,t+1} = \left(A_t + \triangle_t^A\right) x_{k,t} + \left(B_t + \triangle_t^B\right) u_{k,t} \\ y_{k,t+1} = \left(C_{t+1} + \triangle_{t+1}^C\right) x_{k,t+1} \end{cases}, \ t = 0, 1, \cdots, N-1;\ k = 1, 2, \cdots \tag{1}$$

where k denotes the batch number; t represents the time step; N is the batch operation length; $x_{k,t} \in \mathbb{R}^m$, $y_{k,t} \in \mathbb{R}^q$, and $u_{k,t} \in \mathbb{R}^n$ are the system state, output, and input, respectively; $A_t \in \mathbb{R}^{m\times m}$, $B_t \in \mathbb{R}^{m\times n}$, and $C_t \in \mathbb{R}^{q\times m}$ represent the known nominal time-varying model matrices; $\triangle_t^A \in \mathbb{R}^{m\times m}$, $\triangle_t^B \in \mathbb{R}^{m\times n}$, $\triangle_t^C \in \mathbb{R}^{q\times m}$ are the unknown time-varying system uncertainties.

In practice, optimal control design aims to closely follow predefined reference trajectories while minimizing the control signal to reduce energy consumption in batch processes. Thus, the following practical cost function is given for time-varying batch processes:

$$J_{k,t} = \sum_{l=t}^{N-1} \left(e_{k,l+1}^{\top} Q_{l+1} e_{k,l+1} + u_{k,l}^{\top} R_l u_{k,l} \right) \tag{2}$$

with

$$e_{k,t} = y_{r,t} - y_{k,t} \tag{3}$$

where $y_{r,t} \in \mathbb{R}^q$ denotes the reference trajectory, which is repetitive for each batch in the batch processes; $e_{k,t} \in \mathbb{R}^q$ represents the tracking error; $Q_t \succeq 0$ represents the symmetric positive semi-definite weighting matrix; $R_t \succ 0$ denotes symmetric positive definite weighting matrix.

For time-varying batch processes, the control scheme is designed to minimize the specified cost function (2). However, time-varying system dynamics and unknown uncertainties deteriorate control performance and prevent the achievement of the optimal control objectives.

Design Objectives: For the time-varying batch process with unknown time-varying system uncertainties, the optimal control scheme is designed to meet the following objectives:

- The optimal control policy is designed to determine the optimal control signal $\{u_{k,0}^*, u_{k,1}^*, \cdots, u_{k,N-1}^*\}$ to minimize the given cost function $J_{k,t}$.
- The influence of unknown time-varying system uncertainties on the control performance must be eliminated.

3 Control Scheme

This section introduces a novel data-driven Q-learning-based optimal control scheme. Initially, the initial control policy is designed based on the nominal model, excluding time-varying system uncertainties. By using the repetitive nature of batch processes, this mode-based initial control policy is employed to sample operation data from time-varying batch processes. Then, the Q-learning-based optimal control scheme is proposed to optimize the initial control policy from the sampled operation data.

3.1 Model-based Initial Control Policy

To design the model-based optimal control scheme, we first define the following reference trajectory model:

$$\begin{cases} \bar{y}_{r,t} = \sum_{i=1}^{q} y^{i}_{r,t} \\ \bar{y}_{r,t+1} = d_t \bar{y}_{r,t} \\ y_{r,t} = H_t \bar{y}_{r,t} \end{cases} \tag{4}$$

where $y^{i}_{r,t}$ represents the ith scalar of the reference trajectory; $\bar{y}_{r,t} \in \mathbb{R}^1$ is the sum of the reference trajectory; $d_t \in \mathbb{R}^1$ and $H_t \in \mathbb{R}^{q \times 1}$ are the time-varying reference trajectory matrices. Based on the reference trajectory model (4) and the practical batch process (1), the optimal control scheme is derived as follows:

Theorem 1 (Model-based Optimal Control Scheme). For time-varying batch processes with time-varying system uncertainties (1), the optimal control policy π^* and the corresponding minimal cost function $J^*_{k,t}$ are given as follows:

$$\pi^* = \{u^*_{k,0}, u^*_{k,1}, \cdots, u^*_{k,N-1}\} \tag{5}$$

$$u^*_{k,t} = -K^*_t \begin{bmatrix} x_{k,t} \\ \bar{y}_{r,t+1} \end{bmatrix}, \quad J^*_{k,t} = \begin{bmatrix} x_{k,t} \\ \bar{y}_{r,t+1} \end{bmatrix}^{\top} P^*_t \begin{bmatrix} x_{k,t} \\ \bar{y}_{r,t+1} \end{bmatrix} \tag{6}$$

where the optimal control gain matrix K^*_t and the optimal cost function matrix P^*_t are computed as follows: (from $t = N-1$ to $t = 0$):

$$E_t = \begin{bmatrix} A_t + \triangle^A_t & 0 \\ \left(C_{t+1} + \triangle^C_{t+1}\right)\left(A_t + \triangle^A_t\right) & 0 \\ 0 & I \end{bmatrix}, \quad F_t = \begin{bmatrix} B_t + \triangle^B_t \\ \left(C_{t+1} + \triangle^C_{t+1}\right)\left(B_t + \triangle^B_t\right) \\ 0 \end{bmatrix}$$

$$S^*_t = \begin{bmatrix} P^{*,11}_{t+1} & 0 & P^{*,12}_{t+1} d_{t+1} \\ 0 & Q_{t+1} & -Q_{t+1} H_{t+1} \\ d_{t+1} P^{*,21}_{t+1} & -H^{\top}_{t+1} Q_{t+1} & G^*_t \end{bmatrix}, \quad P^*_{t+1} := \begin{bmatrix} P^{*,11}_{t+1} & P^{*,12}_{t+1} \\ P^{*,21}_{t+1} & P^{*,22}_{t+1} \end{bmatrix}$$

$$G^*_t = H^{\top}_{t+1} Q_{t+1} H_{t+1} + d_{t+1} P^{*,22}_{t+1} d_{t+1}, \quad K^*_t = (R_t + F^{\top}_t S^*_t F_t)^{-1} F^{\top}_t S^*_t E_t$$

$$P^*_t = E^{\top}_t \left(S^*_t - S^*_t F_t (R_t + F^{\top}_t S^*_t F_t)^{-1} F^{\top}_t S^*_t \right) E_t$$

Remark 1: At the terminal time $N-1$, there is no cost for the next time. Thus, the optimal cost function J^*_N and the corresponding matrix P^*_N are equal to zero. Additionally, without the reference trajectory $\bar{y}_{r,N+1}$, d_N is set to zero.

Proof of Theorem 1: For the current time t, the given cost function is expressed by:

$$J_{k,t} = e^{\top}_{k,t+1} Q_{t+1} e_{k,t+1} + u^{\top}_{k,t} R_t u_{k,t} + J_{k,t+1} \tag{7}$$

Based on optimality principle [23], the optimal cost function satisfies the following properties:

$$J_{k,t}^{*}=\min\left\{e_{k,t+1}^{\top}Q_{t+1}e_{k,t+1}+u_{k,t}^{\top}R_{t}u_{k,t}\right\}+J_{k,t+1}^{*} \tag{8}$$

Additionally, replacing Eq. (4) and Eq. (6) in Eq. (8) leads to:

$$J_{k,t}^{*}=\min\left\{\begin{bmatrix}x_{k,t+1}\\ y_{k,t+1}\\ \bar{y}_{r,t+1}\end{bmatrix}^{\top}S_{t}^{*}\begin{bmatrix}x_{k,t+1}\\ y_{k,t+1}\\ \bar{y}_{r,t+1}\end{bmatrix}+u_{k,t}^{\top}R_{t}u_{k,t}\right\} \tag{9}$$

Moreover, based on Eq. (1), the augmented dynamic system is formulated:

$$\begin{bmatrix}x_{k,t+1}\\ y_{k,t+1}\\ \bar{y}_{r,t+1}\end{bmatrix}=E_{t}\begin{bmatrix}x_{k,t}\\ \bar{y}_{r,t+1}\end{bmatrix}+F_{t}u_{k,t} \tag{10}$$

Substituting Eq. (10) into Eq. (9) results in a new expression:

$$J_{k,t}^{*}=\min\left\{(E_{t}\begin{bmatrix}x_{k,t}\\ \bar{y}_{r,t+1}\end{bmatrix}+F_{t}u_{k,t})^{\top}S_{t}^{*}(E_{t}\begin{bmatrix}x_{k,t}\\ \bar{y}_{r,t+1}\end{bmatrix}+F_{t}u_{k,t})+u_{k,t}^{\top}R_{t}u_{k,t}\right\} \tag{11}$$

The derivative of the cost function $J_{k,t}^{*}$ with respect to $u_{k,t}$ is obtained:

$$\frac{\partial J_{k,t}^{*}}{\partial u_{k,t}}=2F_{t}^{\top}S_{t}^{*}E_{t}\begin{bmatrix}x_{k,t}\\ \bar{y}_{r,t+1}\end{bmatrix}+2(R_{t}+F_{t}^{\top}S_{t}^{*}F_{t})u_{k,t} \tag{12}$$

By solving Eq. (12) set to zero, the optimal control signal $u_{k,t}^{*}$ is determined:

$$u_{k,t}^{*}=-K_{t}^{*}\begin{bmatrix}x_{k,t}\\ \bar{y}_{r,t+1}\end{bmatrix} \tag{13}$$

By applying Eq. (13) to Eq. (11), the resulting optimal cost function is derived:

$$J_{k,t}^{*}=\begin{bmatrix}x_{k,t}\\ \bar{y}_{r,t+1}\end{bmatrix}^{\top}P_{t}^{*}\begin{bmatrix}x_{k,t}\\ \bar{y}_{r,t+1}\end{bmatrix} \tag{14}$$

Thus, **Theorem 1** is proved. **Fig. 1** depicts the comprehensive structure of the model-based optimal control scheme for time-varying batch processes.

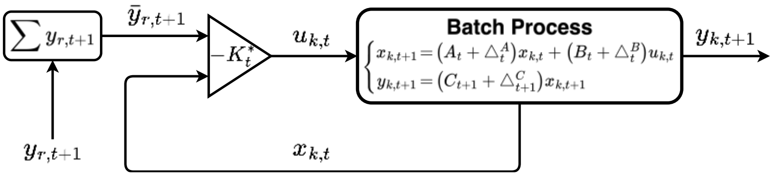
$\sum y_{r,t+1}$
$\bar{y}_{r,t+1}$
$-K_t^*$
$u_{k,t}$
Batch Process
$\begin{cases} x_{k,t+1} = (A_t + \triangle_t^A)x_{k,t} + (B_t + \triangle_t^B)u_{k,t} \\ y_{k,t+1} = (C_{t+1} + \triangle_{t+1}^C)x_{k,t+1} \end{cases}$
$y_{k,t+1}$
$y_{r,t+1}$
$x_{k,t}$

Fig. 1. The overall framework of the model-based optimal control scheme.

3.2 Q-learning-based Optimal Control Scheme

According to **Theorem 1**, the optimal control policy is directly derived when the time-varying system models (1) is fully known. However, the system uncertainties $\triangle_t^A$, $\triangle_t^B$ and $\triangle_t^C$ are not easily determined, which affects the model-based optimal control scheme design. In this subsection, we apply Q-learning to further optimize the initial control policy and eliminate the adverse effects of the unknown time-varying uncertainties. In addition, the following lemma is given to derive the Q-learning-based optimal control scheme.

Lemma 1: For the time-varying batch process, if the control signal satisfies the following structure:

$$u_{k,t} = -K_t \bar{x}_{k,t} \tag{15}$$

at each time *t*, the cost function takes the following quadratic expression:

$$J_{k,t} = \bar{x}_{k,t}^\top P_t \bar{x}_{k,t}, \quad \bar{x}_{k,t} = \begin{bmatrix} x_{k,t} \\ \bar{y}_{r,t+1} \end{bmatrix} \tag{16}$$

where P_t is the symmetric matrix.

We specify the current control policy π for the Q-learning-based control scheme as:

$$\pi = \left\{u_{k,0}^\pi, u_{k,1}^\pi, \cdots, u_{k,N-1}^\pi\right\}, \quad u_{k,t}^\pi = -K_t \bar{x}_{k,t} \tag{17}$$

Then, according to the Q-learning RL and **Lemma 1**, the value function$V_t^\pi(\bar{x}_{k,t})$ under the control policy π at time *t* is expressed by:

$$V_t^\pi(\bar{x}_{k,t}) = \sum_{l=t}^{N-1}\left(e_{k,l+1}^\top Q_{l+1} e_{k,l+1} + u_{k,l}^{\pi\top} R_l u_{k,l}^\pi\right) = \bar{x}_{k,t}^\top P_t^\pi \bar{x}_{k,t} \tag{18}$$

where $u_{k,t}^\pi$ represents the control signal from the control policy π; P_t^π is the cost function matrix corresponding to the control policy π.

Analogous to the value function, the Q-function $Q_t^\pi(\bar{x}_{k,t}, u_{k,t})$ under the current policy π is defined as:

$$Q_t^\pi(\bar{x}_{k,t}, u_{k,t}) = e_{k,t+1}^\top Q_{t+1} e_{k,t+1} + u_{k,t}^\top R_t u_{k,t} + V_{t+1}^\pi(\bar{x}_{k,t+1}) \tag{19}$$

where $u_{k,t}$ is the arbitrary control signal. The subsequent control signals from time $t+1$ to the end of the batch are generated by the current control policy π.

Inserting Eq. (16) into Eq. Eq. (19) yields the following form of the Q-function:

$$Q_t^{\pi}(\bar{x}_{k,t}, u_{k,t}) = \tilde{x}_{k,t+1}^{\top} S_t \tilde{x}_{k,t+1} + u_{k,t}^{\top} R_t u_{k,t} \tag{20}$$

with

$$S_t = \begin{bmatrix} P_{t+1}^{\pi,11} & 0 & P_{t+1}^{\pi,12} d_{t+1} \\ 0 & Q_{t+1} & -Q_{t+1} H_{t+1} \\ d_{t+1} P_{t+1}^{\pi,21} & -H_{t+1}^{\top} Q_{t+1} & H_{t+1}^{\top} Q_{t+1} H_{t+1} + d_{t+1} P_{t+1}^{\pi,22} d_{t+1} \end{bmatrix}$$

$$\tilde{x}_{k,t+1} = \begin{bmatrix} x_{k,t+1} \\ y_{k,t+1} \\ y_{k,t+1}^{r} \end{bmatrix}, \quad P_{t+1}^{\pi} := \begin{bmatrix} P_{t+1}^{\pi,11} & P_{t+1}^{\pi,12} \\ P_{t+1}^{\pi,21} & P_{t+1}^{\pi,22} \end{bmatrix}$$

As indicated in Eq. (10), the relationship between $\tilde{x}_{k,t+1}$ and $\bar{x}_{k,t}$ is represented by:

$$\tilde{x}_{k,t+1} = E_t \bar{x}_{k,t} + F_t u_{k,t} \tag{21}$$

For the off-policy Q-learning RL, we define two control policies: behavior control policy $\bar{\pi}$ and target control policy π (17). The behavior control policy $\bar{\pi}$ generates the behavior control signal $u_{k,t}$ to interact with the batch systems, while the target control policy is employed for policy improvement.

Given the fixed target control policy π, Eq. (21) can be rewritten as:

$$\tilde{x}_{k,t+1}^{\pi} = E_t \bar{x}_{k,t} + F_t u_{k,t}^{\pi} \tag{22}$$

Additionally, the Q-function from the target control policy π and the target control signal $u_{k,t}^{\pi}$ at time t is expressed by:

$$Q_t^{\pi}(\bar{x}_{k,t}, u_{k,t}^{\pi}) = \tilde{x}_{k,t+1}^{\pi\top} S_t \tilde{x}_{k,t+1}^{\pi} + u_{k,t}^{\pi\top} R_t u_{k,t}^{\pi} \tag{23}$$

Incorporating Eq. (21) and Eq. (22) into Eq. (23) gives:

$$\begin{aligned} Q_t^{\pi}(\bar{x}_{k,t}, u_{k,t}^{\pi}) = {} & \tilde{x}_{k,t+1}^{\top} S_t \tilde{x}_{k,t+1} - 2\bar{x}_{k,t}^{\top} E_t^{\top} S_t F_t \left(u_{k,t} - u_{k,t}^{\pi} \right) \\ & - \left(u_{k,t} + u_{k,t}^{\pi} \right)^{\top} F_t^{\top} S_t F_t \left(u_{k,t} - u_{k,t}^{\pi} \right) + u_{k,t}^{\pi\top} R_t u_{k,t}^{\pi} \end{aligned} \tag{24}$$

Based on Eq. (22) and Eq. (23), the Q-function has another expression:

$$Q_t^{\pi}(\bar{x}_{k,t}, u_{k,t}^{\pi}) = \begin{bmatrix} \bar{x}_{k,t} \\ u_{k,t}^{\pi} \end{bmatrix}^{\top} \begin{bmatrix} T_t^{\pi,11} & T_t^{\pi,12} \\ T_t^{\pi,21} & T_t^{\pi,22} \end{bmatrix} \begin{bmatrix} \bar{x}_{k,t} \\ u_{k,t}^{\pi} \end{bmatrix} = \begin{bmatrix} \bar{x}_{k,t} \\ u_{k,t}^{\pi} \end{bmatrix}^{\top} T_t^{\pi} \begin{bmatrix} \bar{x}_{k,t} \\ u_{k,t}^{\pi} \end{bmatrix} \tag{25}$$

with

$$T_t^{\pi,11}=E_t^\top S_t E_t \quad T_t^{\pi,12}=E_t^\top S_t F_t$$
$$T_t^{\pi,21}=F_t^\top S_t E_t \quad T_t^{\pi,22}=R_t+F_t^\top S_t F_t$$

Inserting $T_t^{\pi,12}$, $T_t^{\pi,22}$, and Eq. (25) into Eq. (24) results in the following equation:

$$\bar{x}_{k,t}^\top T_t^{\pi,11}\bar{x}_{k,t}+2\bar{x}_{k,t}^\top T_t^{\pi,12}u_{k,t}+u_{k,t}^\top T_t^{\pi,22}u_{k,t}=\tilde{x}_{k,t+1}^\top S_t\tilde{x}_{k,t+1}+u_{k,t}^\top R_t u_{k,t} \tag{26}$$

To solve the matrix T_t^π, Eq. (26) is reformulated in Kronecker Product form:

$$\psi_t L_t^\pi=w_t \tag{27}$$

with

$$\psi_t=\left[\psi_{t,1}^\top,\psi_{t,2}^\top,\psi_{t,3}^\top\right],\quad L_t^\pi=\begin{bmatrix} vec(T_t^{\pi,11}) \\ vec(T_t^{\pi,12}) \\ vec(T_t^{\pi,22}) \end{bmatrix},\quad w_t=\tilde{x}_{k,t+1}^\top S_t\tilde{x}_{k,t+1}+u_{k,t}^\top R_t u_{k,t}$$
$$\psi_{t,1}=\bar{x}_{k,t}\otimes\bar{x}_{k,t},\psi_{t,2}=2\bar{x}_{k,t}\otimes u_{k,t},\psi_{t,3}=u_{k,t}\otimes u_{k,t}$$

where $\otimes$ is the Kronecker Product; The function $vec(\cdot)$ denotes the vectorization of the matrix. By using the repetitive nature of the batch processes, the behavior control policy $\bar{\pi}$ is utilized in the controlled batch systems to sample the process data. With r sets of the sampled process data, Eq. (27) is formulated as follows:

$$\Psi_t L_t^\pi=W_t \tag{28}$$

with

$$\Psi_t=\left[\psi_t^{1\top},\psi_t^{2\top},\cdots,\psi_t^{r\top}\right]^\top,\quad W_t=\left[w_t^1,w_t^2,\cdots,w_t^r\right]^\top$$

where the superscript indicates the rth process data. Based on Eq. (28) and the least squares method [24], the vector L_t^π is computed. Subsequently, the matrix T_t^π is reconstructed from the vector L_t^π as follows:

$$T_t^\pi=vec^{-1}(L_t^\pi) \tag{29}$$

Remark 2: It contains $\alpha=0.5*(n+m+1)(n+m+2)$ independent parameters to be estimated for the symmetric matrix T_t^π. By the least squares estimation framework, the number of data samples r must satisfy the condition $r\geq\alpha$ to ensure parameter identifiability.

The optimal control policy is derived via policy evaluation and policy improvement in the Q-learning-based optimal control scheme.

Policy Evaluation: The Q-function $Q_t^\pi(\bar{x}_{k,t},u_{k,t}^\pi)$ is obtained by the reverse recursive framework (from $t=N-1$ to $t=0$):

```
Algorithm 1 Q-learning-based Optimal Control Scheme
Input the known nominal system matrices $A_t B_t C_t$.
Give the reference trajectory $y_{r,t}$ and compute matrices $d_t$ and $H_t$.
Set time-varying cost function weighting matrices $Q_t$ and $R_t$.
Compute the initial control policy based on Theorem 1:
    $\pi^0=\left\{u^{\pi^0}_{k,0}, u^{\pi^0}_{k,1}, \cdots, u^{\pi^0}_{k,N-1}\right\}, \quad u^{\pi^0}_{k,t}=-K^0_t\bar{x}_{k,t}$
Using the initial control policy plus an exploration noise $\xi$ to sample $r$ batches
    data and save data $\left(\bar{x}_{k,t}, u_{k,t}, \tilde{\bar{x}}_{k,t+1}\right)$ in the buffer $\mathbb{D}$.
Set the iteration number $i_{max}$ and the iteration termination condition $\varepsilon$.
for $i=0,1,\cdots,i_{max}$ do
    for $t=N-1,\cdots,1,0$ do
        Policy Evaluation:
        The sampled operation data from buffer $\mathbb{D}$ is used to construct $\Psi_t$ and $W_t$.
        Determine the matrix $T^{\pi^i}_t$ as follows:
            $\Psi_t L^{\pi^i}_t = W_t$        (30)
        Calculate the matrix $P^{\pi^i}_t$ for policy evaluation based on Eq. (33).
        Policy Improvement:
        Improve the control gain matrix $K^{i+1}_t$ based on the obtained matrix $T^{\pi^i}_t$:
            $K^{i+1}_t=\left(T^{\pi^i,22}_t\right)^{-1}T^{\pi^i,21}_t$
    end for
    The enhanced control policy $\pi^{i+1}$ is formulated below:
        $\pi^{i+1}=\left\{u^{\pi^{i+1}}_{k,0}, u^{\pi^{i+1}}_{k,1}, \cdots, u^{\pi^{i+1}}_{k,N-1}\right\}, \quad u^{\pi^{i+1}}_{k,t}=-K^{i+1}_t\bar{x}_{k,t}$
    When conditions $\left\|K^{i+1}_t-K^i_t\right\|<\varepsilon$ are satisfied, the iteration stops.
end for
Finally, the Q-learning-based optimal control policy is applied to the time-varying batch process.
```

$$Q^{\pi}_t(\bar{x}_{k,t}, u^{\pi}_{k,t}) = \begin{bmatrix} \bar{x}_{k,t} \\ u^{\pi}_{k,t} \end{bmatrix}^{\top} T^{\pi}_t \begin{bmatrix} \bar{x}_{k,t} \\ u^{\pi}_{k,t} \end{bmatrix} \tag{31}$$

As stated by Eq. (28) and Eq. (29), the matrix T^{π}_t is determined. However, the matrix P^{π}_{t+1} must be determined at time $t+1$. Built upon the Q-learning RL algorithm, the value function and Q-function have the following property:

$$V^{\pi}_t(\bar{x}_{k,t}) = Q^{\pi}_t(\bar{x}_{k,t}, u^{\pi}_{k,t}) \tag{32}$$

Inserting Eq. (17), Eq. (18), and Eq. (25) into Eq. (32) yields the following expression:

$$P_t^{\pi}=\begin{bmatrix} I \\ -K_t \end{bmatrix}^{\top} T_t^{\pi} \begin{bmatrix} I \\ -K_t \end{bmatrix} \tag{33}$$

Policy Improvement: Based on the obtained Q-function $Q_t^{\pi}(\bar{x}_{k,t}, u_{k,t}^{\pi})$, the current control policy π is improved by:

$$\tilde{K}_t = \underset{K_t}{\operatorname{argmin}} Q_t^{\pi}(\bar{x}_{k,t}, u_{k,t}^{\pi}) \tag{34}$$

According to Eq. (31), the Q-function $Q_t^{\pi}(\bar{x}_{k,t}, u_{k,t}^{\pi})$ is differentiated on the target control signal $u_{k,t}^{\pi}$ and setting it to zero, the improved target control signal is determined by:

$$u_{k,t}^{\tilde{\pi}} = -\tilde{K}_t \bar{x}_{k,t}, \quad \tilde{K}_t = \left(T_t^{\pi,22}\right)^{-1} T_t^{\pi,21} \tag{35}$$

The improved control policy $\tilde{\pi}$ is noted by:

$$\tilde{\pi} = \left\{u_{k,0}^{\tilde{\pi}}, u_{k,1}^{\tilde{\pi}}, \cdots, u_{k,N-1}^{\tilde{\pi}}\right\} \tag{36}$$

The current control policy π is undoubtedly improved by the Q-learning-based optimal control scheme. **Algorithm 1** outlines the Q-learning-based optimal control framework. The overall structure of the Q-learning-based optimal control scheme is depicted in

Fig. 2.

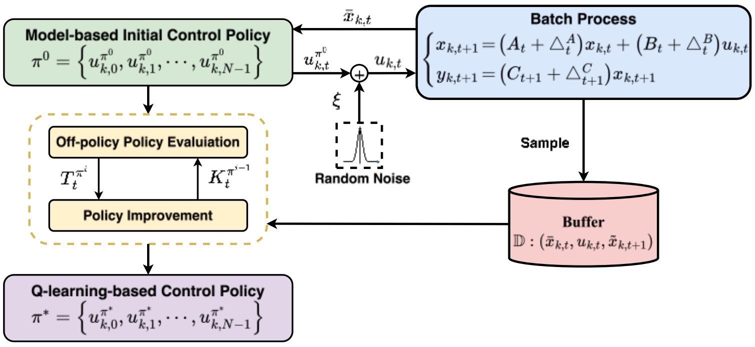
Model-based Initial Control Policy
$\pi^0 = \left\{u_{k,0}^{\pi^0}, u_{k,1}^{\pi^0}, \cdots, u_{k,N-1}^{\pi^0}\right\}$
$\bar{x}_{k,t}$
$u_{k,t}^{\pi^0}$
$u_{k,t}$
Batch Process
$\begin{cases} x_{k,t+1} = (A_t + \triangle_t^A)x_{k,t} + (B_t + \triangle_t^B)u_{k,t} \\ y_{k,t+1} = (C_{t+1} + \triangle_{t+1}^C)x_{k,t+1} \end{cases}$
ξ
Random Noise
Sample
Off-policy Policy Evaluation
$T_t^{\pi^i}$
$K_t^{\pi^{i-1}}$
Policy Improvement
Buffer
$\mathbb{D} : (\bar{x}_{k,t}, u_{k,t}, \tilde{x}_{k,t+1})$
Q-learning-based Control Policy
$\pi^* = \left\{u_{k,0}^{\pi^*}, u_{k,1}^{\pi^*}, \cdots, u_{k,N-1}^{\pi^*}\right\}$

Fig. 2. The structure of the proposed Q-learning-based optimal control scheme.

4 Numerical Simulations

To validate the efficacy and control performance of the Q-learning-based optimal control scheme, we conduct simulations on a numerical linear MIMO time-varying batch system and a time-varying injection molding process.

4.1 Linear MIMO Batch System

The numerical linear MIMO time-varying batch system is considered and modeled by:

$$\begin{cases} x_{k,t+1} = \left(A_t + \triangle_t^A\right) x_{k,t} + \left(B_t + \triangle_t^B\right) u_{k,t} \\ y_{k,t+1} = \left(C_{t+1} + \triangle_{t+1}^C\right) x_{k,t+1} \end{cases} \tag{37}$$

with

$$A_t \equiv \begin{bmatrix} 0 & 1 & -1 \\ -0.6 & -1.1 & 0.6 \\ -0.4 & -1.2 & 0.5 \end{bmatrix}, B_t \equiv \begin{bmatrix} 1 & 2 \\ 10 & 6 \\ 5 & 3 \end{bmatrix}, C_t \equiv \begin{bmatrix} 1 & 1.5 & 1 \\ 1 & 0.5 & 0.5 \end{bmatrix}$$

$$\triangle_t^A = 0.1 A_t \sin(t), \ \triangle_t^B = 0.2 B_t \cos(t), \ \triangle_t^C = 0.05 C_t \sin(2t)$$

where A_t, B_t, and C_t are known system matrices of the linear MIMO batch system; $\triangle_t^A$, $\triangle_t^B$, and $\triangle_t^C$ represent unknown time-varying system uncertainties; The fixed time duration per batch N is 100; All entries of the initial system state $x_{k,0}$ is generated randomly between -5 and 5. Additionally, the desired trajectory is given by:

$$y_{r,t} = \begin{cases} \begin{bmatrix} 10 & 20 \end{bmatrix}^\top & , 0 \le t \le 50 \\ \begin{bmatrix} 20 & 30 \end{bmatrix}^\top & , 50 < t \le 100 \end{cases} \tag{38}$$

First, for the Q-learning-based optimal control scheme, the time-varying reference trajectory model matrices d_t and H_t are computed based on the given desired trajectory. According to mode-based optimal control scheme (**Theorem 1**), the initial model-based control policy is then designed:

$$\pi^0 = \left\{ u_{k,0}^{\pi^0}, u_{k,1}^{\pi^0}, \cdots, u_{k,N-1}^{\pi^0} \right\}, \quad u_{k,t}^{\pi^0} = -K_t^0 \bar{x}_{k,t}$$

where the time-varying cost function weighting matrices Q_t and R_t are set as follows:

$$Q_t \equiv \begin{bmatrix} 200 & 0 \\ 0 & 100 \end{bmatrix}, \quad R_t \equiv \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}$$

In addition, the exploration batch $r = 25$ is specified. The initial control policy π^0 adds a random exploration noise ξ to constitute the behavior control signal $u_{k,t}$ to sample the process data:

$$u_{k,t} = u^{\pi^0}_{k,t} + \xi \tag{39}$$

where ξ represents the random exploration noise between -1 and 1.

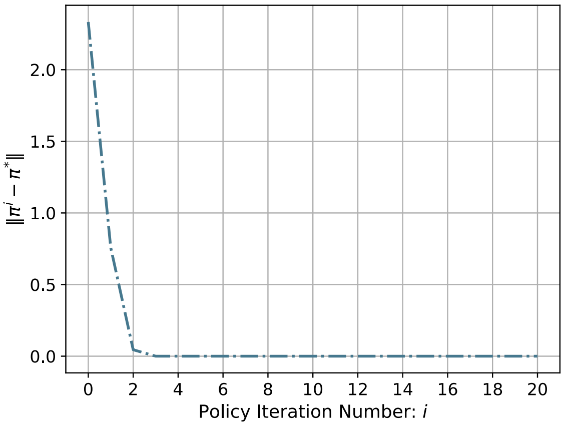

Fig. 3. The convergence curve of the policy iteration.

After data collection, the proposed Q-learning-based optimal control scheme **Algorithm 1** is employed to improve the initial control policy π^0. In the process of policy iteration, the specific iterative parameters are set with the maximum iteration number $i_{max} = 20$ and the iteration termination condition $\varepsilon = 0$. **Fig. 3** illustrates that the control policy progressively converges to the optimal solution through the policy iteration by the Q-learning-based optimal control scheme.

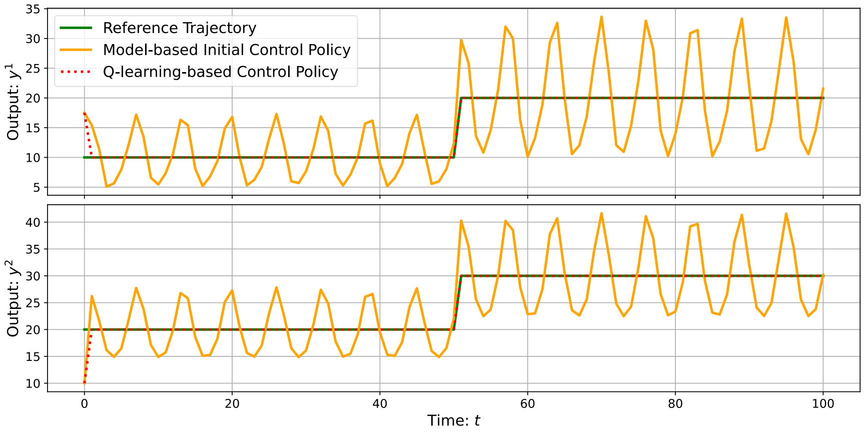

Fig. 4. The output response of the MIMO batch .

Finally, after 25 batches of data sampling and 20 off-policy policy iterations, the trained Q-learning-based control policy is applied to control the MIMO batch system and compare control performance with the model-based initial control policy. As shown in **Fig. 4**, the output response under the model-based initial control policy exhibits significant deviation from the reference trajectory, primarily due to its inability to compensate for control errors induced by time-varying system uncertainties. This result highlights the substantial negative impact of time-varying uncertainties on control performance, thereby limiting the practical applicability of such model-based approaches in real-world systems. Conversely, the output response under the Q-learning control policy closely follows the trajectory reference, demonstrating that the proposed Q-learning-based optimal control scheme achieves superior tracking performance. For an accurate comparison of the control performance, the RMSE (the root-mean-square error) and SAE (the sum of absolute error) of the tracking error for the entire batch are listed in **Table 1**. As illustrated in **Table 1**, the RMSE and the SAE corresponding to the Q-learning control policy are much lower than the initial control policy. This experimental simulation shows that the Q-learning-based optimal control scheme is an effective control method for time-varying batch processes.

Table 1. RMSE and SAE of the initial control policy and Q-learning-based control scheme.

Algorithm	**RMSE**	**SAE**
Initial Control Policy	8.53	1032.44
Q-learning-based Control Policy	0.008	1.017

4.2 Injection Molding

Injection molding is considered a fundamental batch process in chemical engineering. Typically, the injection molding is modeled as a time-varying batch process with unknown, time-dependent uncertainties [19]:

$$\begin{cases} x_{k,t+1} = \left(A_t + \triangle_t^A\right) x_{k,t} + \left(B_t + \triangle_t^B\right) u_{k,t} \\ y_{k,t+1} = \left(C_{t+1} + \triangle_{t+1}^C\right) x_{k,t+1} \end{cases} \tag{40}$$

with

$$A_t = \begin{bmatrix} 1.607 & 1 \\ -0.6086 & 0 \end{bmatrix} \sigma_t^A, B_t = \begin{bmatrix} 1.239 \\ -0.9283 \end{bmatrix} \sigma_t^B, C_t = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

$$\triangle_t^A = \begin{bmatrix} 0.0604 & -0.0204 \\ -0.0204 & 0 \end{bmatrix} \delta_t^A, \triangle_t^B = \begin{bmatrix} 0.062 \\ -0.0464 \end{bmatrix} \delta_t^B, \triangle_t^C = \begin{bmatrix} 0.01 & -0.01 \end{bmatrix} \delta_t^C$$

$$\sigma_t^A = 0.5 + 0.2e^{0.05t}, \sigma_t^B = 1 + 0.1e^{0.05t}, \delta_t^A = e^{0.05t}, \delta_t^B = \delta_t^C = \sin(t)$$

where A_t, B_t, and C_t are known model matrices of the injection molding; $\triangle_t^A$, $\triangle_t^B$, and $\triangle_t^C$ represent unknown time-varying uncertainties; The operation time N of each batch is 200; The initial state $x_{k,0}$ is generated randomly between $\begin{bmatrix} -50 & -50 \end{bmatrix}^\top$ and

$\begin{bmatrix} 50 & 50 \end{bmatrix}^\top$. According to the practical applications, the following trajectory profile is given:

$$y_{r,t} = \begin{cases} 200 & ,0 \le t \le 100 \\ 200 + 5(t - 100) & ,100 < t \le 120 \\ 300 & ,120 < t \le 200 \end{cases} \tag{41}$$

Analogously, without system uncertainties, **Theorem 1** is used to design the initial model-based control policy:

$$\pi^0 = \left\{ u^{\pi^0}_{k,0}, u^{\pi^0}_{k,1}, \cdots, u^{\pi^0}_{k,N-1} \right\}, \quad u^{\pi^0}_{k,t} = -K^0_t \bar{x}_{k,t}$$

where the time-varying cost function weighting matrices $Q_t \equiv 100$, $R_t \equiv 0.1$, and the exploration batches $r = 10$ are specified. Meanwhile, the control signals $u_{k,t}$ from the initial control policy π^0 adds a random exploration noise to control the injection molding process:

$$u_{k,t} = u^{\pi^0}_{k,t} + 5\xi \tag{42}$$

After sampling the data, the Q-learning-based optimal control scheme **Algorithm 1** is used to improve the initial model-based control policy iteratively. The policy iteration number i_{max} and the termination condition ε are set as 15 and 0, respectively. **Fig. 5** shows that the initial control policy π^0 significantly differs from the optimal control policy π^*. Through the policy iteration, the control policy progressively converges into the optimal control policy π^*.

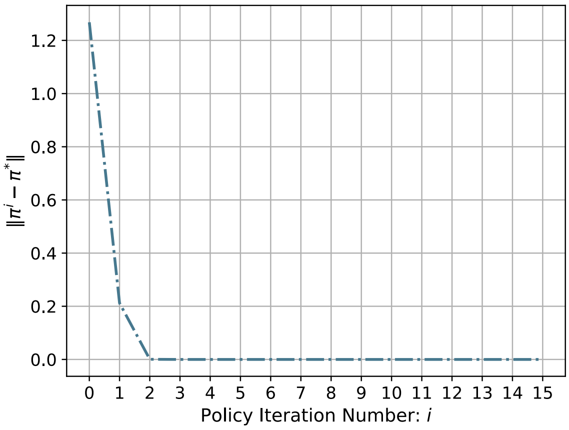

Fig. 5. The convergence curve of the policy iteration.

After the 10 batches of data sampling and 15 policy improvements, the Q-learning-based control policy is tested for the control performance in the time-varying injection molding process. As illustrated in **Fig. 6**, the output response governed by the model-based initial control policy exhibits pronounced deviation from the reference trajectory. This deviation stems from the policy's inability to compensate for dynamic perturbations introduced by time-varying system uncertainties. The result quantitatively demonstrates the detrimental effect of uncertainties on control performance. In contrast, the output response under the Q-learning control policy is very close to the trajectory reference, showing the superior tracking performance of the proposed Q-learning-based optimal control scheme. In addition, we compare the control performance of the Q-learning-based control scheme with the initial control policy and the model-based PI-type indirect ILC [19]. After 40 batch iterations, the PI-based indirect ILC control scheme exhibits better control performance but remains less effective than the Q-learning-based optimal control scheme. Additionally, we compute the RMSE and SAE of the three control schemes in **Table 2** to compare the control performance more precisely. Both the RMSE and the SAE of the Q-learning-based control policy are lower than that of the PI-based indirect ILC even after 40 iterations, which indicates the optimality of the proposed Q-learning-based optimal control scheme.

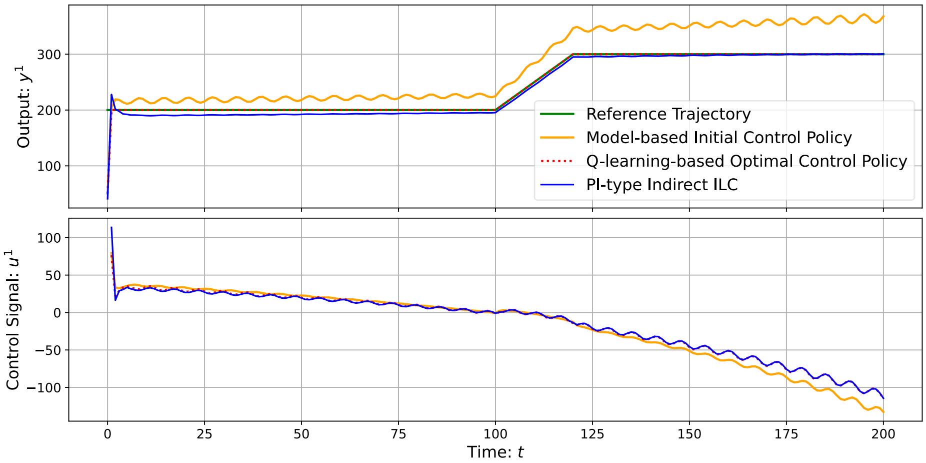

Fig. 6. The injection molding process's output response and control signal.

Table 2. RMSE and SAE of the three control schemes.

Algorithm	**RMSE**	**SAE**
Initial Control Policy	39.29	7090.62
PI-type Indirect ILC after 40 Batch	6.124	1017.30
Q-learning-based Control Policy	0.013	1.57

In summary, the simulations in the numerical MIMO batch system and the time-varying injection molding process demonstrate that the proposed Q-learning-based optimal control scheme is a universal control solution for time-varying batch processes with time-varying system uncertainties.

5 Conclusions

Given the widespread application of batch processes in modern industry, we propose a novel data-driven Q-learning-based optimal control scheme for time-varying batch processes with time-varying uncertainties in this paper. The nominal model-based initial control policy is derived based on the principle of optimality and dynamic programming. However, this initial control policy is non-optimal due to the unknown time-varying system uncertainties. By leveraging the repeatability of the batch processes, the non-optimal control policy is utilized to sample the operation data from the time-varying batch process. Then, the Q-learning-based optimal control scheme is developed to optimize the initial control policy to eliminate the impact of the time-varying system uncertainties. Finally, the numerical simulations on the MIMO time-varying batch system and the injection molding process illustrate the effectiveness and advantages of the proposed Q-learning-based optimal control scheme.

References

1. Pierallini M, Angelini F, Bicchi A, et al. Swing-up of underactuated compliant arms via iterative learning control[J]. IEEE Robotics and Automation Letters, 2022, 7(2): 3186-3193.
2. Steinhauser A, Swevers J. An efficient iterative learning approach to time-optimal path tracking for industrial robots[J]. IEEE Transactions on Industrial Informatics, 2018, 14(11): 5200-5207.
3. Joshi T, Makker S, Kodamana H, et al. Twin actor twin delayed deep deterministic policy gradient (TATD3) learning for batch process control[J]. Computers & Chemical Engineering, 2021, 155: 107527.
4. Liu J, Liu T, Chen J, et al. Data-driven modeling of product crystal size distribution and optimal input design for batch cooling crystallization processes[J]. Journal of Process Control, 2020, 96: 1-14.
5. Li M, Chen T, Cheng R, et al. Dual-loop iterative learning control with application to an ultraprecision wafer stage[J]. IEEE Transactions on Industrial Electronics, 2021, 69(11): 11590-11599.
6. Xu J X. A survey on iterative learning control for nonlinear systems[J]. International Journal of Control, 2011, 84(7): 1275-1294.
7. Lee J H, Lee K S. Iterative learning control applied to batch processes: An overview[J]. Control Engineering Practice, 2007, 15(10): 1306-1318.
8. Bradford E, Imsland L, Zhang D, et al. Stochastic data-driven model predictive control using gaussian processes[J]. Computers & Chemical Engineering, 2020, 139: 106844.
9. Li H, Wang S, Shi H, et al. Two-dimensional iterative learning robust asynchronous switching predictive control for multiphase batch processes with time-varying delays[J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2023, 53(10): 6488-6502.
10. Yoo H, Byun H E, Han D, et al. Reinforcement learning for batch process control: Review and perspectives[J]. Annual Reviews in Control, 2021, 52: 108-119.
11. Chi R, Hou Z, Jin S, et al. An improved data-driven point-to-point ILC using additional online control inputs with experimental verification[J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2017, 49(4): 687-696.
12. Ma Y, Zhu W, Benton M G, et al. Continuous control of a polymerization system with deep reinforcement learning[J]. Journal of Process Control, 2019, 75: 40-47.

13. Chen H, Xu R, Tang Y, et al. Fault-tolerant iterative learning control for batch processes with time-varying state delays and uncertainties[J]. International Journal of Systems Science, 2023, 54(11): 2423-2441.
14. Lu J, Cao Z, Hu Q, et al. Optimal iterative learning control for batch processes in the presence of time-varying dynamics[J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2020, 52(1): 680-692.
15. Owens D H, Owens D H. Norm optimal iterative learning control[J]. Iterative Learning Control: An Optimization Paradigm, 2016: 233-276.
16. Sebastian G, Tan Y, Oetomo D. Convergence analysis of feedback-based iterative learning control with input saturation[J]. Automatica, 2019, 101: 44-52.
17. Tanaskovic M, Fagiano L, Gligorovski V. Adaptive model predictive control for linear time varying MIMO systems[J]. Automatica, 2019, 105: 237-245.
18. Son T D, Pipeleers G, Swevers J. Robust monotonic convergent iterative learning control[J]. IEEE Transactions on Automatic Control, 2015, 61(4): 1063-1068.
19. Hao S, Liu T, Gao F. PI based indirect-type iterative learning control for batch processes with time-varying uncertainties: A 2D FM model based approach[J]. Journal of Process Control, 2019, 78: 57-67.
20. Zhang R, Wu S, Cao Z, et al. A systematic min–max optimization design of constrained model predictive tracking control for industrial processes against uncertainty[J]. IEEE transactions on control systems technology, 2017, 26(6): 2157-2164.
21. Lopez V G, Alsalti M, Müller M A. Efficient off-policy Q-learning for data-based discrete-time LQR problems[J]. IEEE Transactions on Automatic Control, 2023, 68(5): 2922-2933.
22. Shi H, Yang C, Jiang X, et al. Novel two-dimensional off-policy Q-learning method for output feedback optimal tracking control of batch process with unknown dynamics[J]. Journal of Process Control, 2022, 113: 29-41.
23. Lewis F L, Vrabie D, Syrmos V L. Optimal control[M]. John Wiley & Sons, 2012.
24. Branch M A, Coleman T F, Li Y. A subspace, interior, and conjugate gradient method for large-scale bound-constrained minimization problems[J]. SIAM Journal on Scientific Computing, 1999, 21(1): 1-23.