LYAPUNOV-BASED MODEL PREDICTIVE CONTROL FOR CONTINUOUS STIRRED-TANK REACTORS USING PHYSICS-INFORMED NEURAL NETWORKS AND DISTURBANCE OBSERVER

Peng LU[1], Chenghu JING[*2], Yafeng LIU[2]

[1]School of International Education, Henan University of Technology, Zhengzhou 450001, China

[*2]School of Mechanical and Electronic Engineering, Henan University of Technology, Zhengzhou 450001, China,

[*] Chenghu JING; E-mail: chhjing@haut.edu.cn

To address modeling complexity and poor control performance of traditional methods for continuous stirred-tank reactors subject to severe nonlinearity, and multivariable coupling, this paper proposes a Lyapunov-based model predictive control strategy integrating physics-informed neural networks and disturbance observers. First, a physics-informed neural network incorporates physical equation residuals into network training, establishing a high-precision discrete data-driven dynamic model without complex mechanistic modeling. Second, to prevent potential divergence of purely data-driven model predictive control in unstable processes, a Lyapunov constraint is incorporated into the optimization scheme to guarantee closed-loop stability. Additionally, a proportional-integral disturbance observer estimates lumped process disturbances online for feedforward and integral compensation. Numerical simulations demonstrate that the physics-informed neural network achieves high prediction accuracy. Under nominal and disturbed conditions, the proposed control strategy achieves rapid and smooth setpoint tracking, eliminates steady-state errors, and demonstrates superior dynamic performance and robustness.

Key words*: continuous stirred-tank reactor, model predictive control, physics-informed neural network, disturbance observer, nonlinear process control*

1. Introduction

Continuous stirred-tank reactors (CSTRs) are fundamental in chemical, pharmaceutical, and refining industries, yet their intrinsic strong exothermicity, severe nonlinearity, multivariable coupling, and harsh operating conditions lead to extremely complex dynamics [1,2]. The reactor state critically determines product quality, yield, and safety, making efficient and reliable control a major engineering challenge [3,4].Various control strategies have been developed. Classical PID control, though simple, suffers from difficult tuning and inadequate global performance for strongly nonlinear systems with delays [5]. Intelligent methods—fuzzy logic [6], neural networks [7], and sliding mode control [8]—handle nonlinearities and uncertainties effectively but rely heavily on expert knowledge or accurate models and lack explicit constraint handling.

Model Predictive Control (MPC), as a model-based optimal approach, excels at handling multivariable coupling, physical constraints, and receding-horizon optimization, becoming a mainstream industrial paradigm [9,10]. Conventional MPC relies on first-principle mechanistic models [11], whose development is costly and difficult for complex CSTRs. To address model uncertainty and external disturbances, robust nonlinear MPC variants have been proposed: multi-stage NMPC uses scenario trees [12], tube-based MPC constrains trajectory deviations [13,14], and minimax MPC optimizes worst-case performance [15]. However, these methods face practical limitations—scenario-tree growth, complicated tube construction, and NP-hard optimization, respectively. Lyapunov-based MPC (LBMPC) offers a more streamlined framework by explicitly embedding a Lyapunov decay constraint, guaranteeing closed-loop stability and asymptotic convergence with much lower computational load [17].Although purely data-driven models like ANNs and LSTMs are widely used in CSTR-MPC [18–20], they may violate physical laws under sparse data or shifting conditions [21].. Physics-informed neural networks (PINNs) embed physical residual terms into the loss function, combining prior knowledge with data learning to ensure both accuracy and physical consistency [22,23]. Integrating PINNs with MPC retains constraint handling while avoiding unphysical predictions. Nevertheless, open-loop PINN predictions alone cannot guarantee closed-loop asymptotic stability. This paper therefore incorporates Lyapunov stability theory into MPC, imposing a decreasing Lyapunov constraint to theoretically ensure stability for CSTRs with unstable equilibria. Furthermore, conventional LBMPC is susceptible to steady-state errors caused by model mismatch and external disturbances. A disturbance observer (DO) is thus introduced to provide online estimation and feed-forward compensation of lumped disturbances, eliminating steady-state offsets and enhancing robustness.

Focusing on the CSTR, we design a data-driven control framework that simultaneously ensures theoretical stability and engineering robustness. A high-precision discrete dynamic model is first established via PINN; then a PINN-DO-LBMPC controller is developed by integrating Lyapunov constraints and the disturbance observer. Numerical simulations demonstrate excellent set-point tracking and convergence under nominal conditions, as well as superior disturbance rejection under severe disturbances and measurement noise, offering a safe and efficient solution for nonlinear process control. The remainder of this paper is organized as follows: Section 2 presents the CSTR dynamic model. Section 3 describes the PINN architecture, training, and multi-step prediction. Section 4 details the controller design, including the PI observer and Lyapunov-constrained MPC. Section 5 presents simulation results, and Section 6 concludes.

2. System Dynamic Model

This paper considers a typical continuous stirred-tank reactor in which the following irreversible exothermic reaction takes place: $A \xrightarrow{k} B$. The reaction is of second order, and the reaction rate follows the Arrhenius law. Based on material balance and energy balance, the dynamic behavior of the system can be described by the following nonlinear differential equations[24]:

$$\begin{aligned} \frac{dC_A}{dt} &= \frac{F}{V}\left(C_{A0} - C_A\right) - k_0 e^{-E/(RT)} C_A^2 \\ \frac{dT}{dt} &= \frac{F}{V}\left(T_0 - T\right) + \frac{\left(-\Delta \mathrm{H}\right)}{\rho c_p} k_0 e^{-E/(RT)} C_A^2 + \frac{Q}{\rho c_p V} \end{aligned} \tag{1}$$

The physical meanings and values of the variables and parameters are listed in tab. 1[25]. The system has an open-loop unstable equilibrium point, whose steady-state values are given by: $C_{As} = 1.9537$ kmol/m^{3}and $T_s = 401.8727$ K. The corresponding steady-state values of feed concentration and heat input are: $C_{Aos} = 4$ kmol/m^3 and $Q_s = 0$ kJ/h.

To facilitate the controller design, the state deviations and control deviations are defined as follows:

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} C_A - C_{As} \\ T - T_s \end{bmatrix}, \boldsymbol{u} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} C_{A0} - C_{A0s} \\ Q - Q_s \end{bmatrix} \tag{2}$$

By rewriting eq. (1) in deviation form, the nonlinear dynamics of the system near the equilibrium point can be obtained. In this paper, it is assumed that the materials inside the reactor are perfectly mixed, the physical parameters are constant, and the feed flow rate and feed temperature are fixed. These assumptions form the basis for the subsequent simplification of the mechanistic model and data generation. Disturbances including unmodeled dynamics, parameter uncertainties, and external disturbances are considered. The actual system with unknown disturbances is represented as :

$$x(k+1) = f_{true}\left(x(k), u(k)\right) + d(k) \tag{3}$$

where, $d(k) \in \mathbb{R}^2$is the equivalent total perturbation (including model error and external interference). $f_{true}\left(\boldsymbol{x}(k), \boldsymbol{u}(k)\right)$ is a system model, and it is proposed to use a neural network to approximate:

$$f_{\text{true}}(\boldsymbol{x}(k), \boldsymbol{u}(k)) \approx f_{\text{NN}}(\boldsymbol{x}(k), \boldsymbol{u}(k)) \tag{4}$$

where, $f_{\text{NN}}(\boldsymbol{x}(k), \boldsymbol{u}(k))$ is the trained neural network model.

The control objective of this system is to regulate the feed concentration C_{A0} and heat input Q such that the reactor concentration C_A and temperature T track the setpoints rapidly and smoothly, while ensuring that the control inputs stay within their allowable physical constraints.

Table 1 Variables and parameters of CSTR model

Description	Value	Description	Value
Feed flow rate F	5 m³/h	Universal gas constant R	8.314 J/(mol·K)
Reactor volume V	1 m³	Feed temperature T_0	300 K
Pre-exponential factor k_0	8.46×10⁶ m³/kmol hr	Enthalpy of reaction ΔH	-11500 J/mol
Activation energy E	50000 J/mol	Specific heat capacity c_p	0.231 kJ/(kg·K)

3. PINN Model Construction

To overcome the difficulties of mechanistic modeling while fully leveraging known physical laws, this paper employs PINN to establish a discrete-time prediction model for the CSTR. Core idea of PINN is to incorporate physical equations as prior knowledge into network training, enabling the model to be data-driven while satisfying physical constraints.

3.1. Data Generation and Process

To train the network, open-loop simulations are performed on the mechanistic model of eq.(1) to generate input-output samples. The sampling period is $\Delta t = 0.01$h. However, the training data for the PINN includes not only the terminal states but also intermediate states to learn the complete dynamic mapping. Specifically, uniform sampling is conducted within the feasible ranges of the control inputs

u_1 and u_2 (15 points each) and within a large range of the initial states $x_1(0)$ and $x_2(0)$. All sampled initial points are constrained within the stable operation region of the reactor to avoid unphysical dynamic trajectories. For each group of initial state and control input, numerical integration is performed with a tiny integration step $h_c = 10^{-4}$h. The simulation runs until the total simulation duration equals one control sampling cycle Δt. The system states captured at intermediate time points $t \in \{0.002, 0.005, 0.008, 0.01\}$h are recorded as the target outputs. This results in approximately 80,000 samples, which are randomly divided into a training set (80%) and a validation set (20%). The sampling ranges for the control inputs and initial states are specified as follows: $u_1 \in [-3.5, 3.5]$ kmol/m³, $u_2 \in [-5 \times 10^5, 5 \times 10^5]$ kJ/h, $x_1(0) \in [-2.5, 2.5]$ kmol/m³, and $x_2(0) \in [-120, 120]$ K. These ranges were chosen to cover the physically feasible operating region while respecting the stability boundaries of the CSTR system. To eliminate magnitude scaling, inputs and outputs are normalized via $\widetilde{\boldsymbol{X}} = (\boldsymbol{X} - \boldsymbol{\mu_X})/\boldsymbol{\sigma_X}$ and $\widetilde{\boldsymbol{Y}} = (\boldsymbol{Y} - \boldsymbol{\mu_Y})/\boldsymbol{\sigma_Y}$, with parameters: $\boldsymbol{\mu}_X = [0, -0.02, 0.58, 0,\ 5.2 \times 10^{-13}]^T$, $\boldsymbol{\sigma}_X = [0.0037, 0.87, 39.45, 2.16, 3.09 \times 10^5]^T$, $\boldsymbol{\mu}_Y = [0.01, 0.31]^T$, and $\boldsymbol{\sigma}_Y = [0.86, 40.30]^T$.

3.2. Network Architecture and Training Process

The PINN designed in this paper adopts a fully connected feedforward neural network. The input vector contains 5 features: the prediction time interval $\tau \in (0\ , \Delta t]$, the current concentration deviation $x_1(k)$, $x_2(k)$, and control inputs $u_1(k)$, $u_2(k)$. When setting $\tau = \Delta t$ (the control sampling period), the network predicts the state deviation vector at the next sampling instant $k + 1$:

$$[\hat{x}_1(k+1), \hat{x}_2(k+1)]^T = f_{NN}(\tau, x_1(k), x_2(k), u_1(k), u_2(k)) \tag{5}$$

The output is the predicted state deviation vector $\hat{\boldsymbol{x}}(k+1) = [\hat{x}_1(k+1), \hat{x}_2(k+1)]$. The network architecture consists of an input layer (5 nodes), three hidden layers (64, 64, and 32 nodes, respectively, all utilizing the tanh activation function), and an output layer (2 nodes with linear activation).

The training employs the mean squared error (MSE) as the loss function, with the Adam optimizer, a learning rate of 0.01, a batch size of 512, and 50 training epochs. During training, the loss reduction is monitored, and the performance is evaluated on the validation set. The loss function used for PINN training explicitly consists of two components: the data loss and the physics residual loss. The total loss is defined as $\mathcal{L} = \mathcal{L}_{\text{data}} + \lambda \mathcal{L}_{\text{physics}}$, where the data loss is the mean squared error (MSE) between the predicted and true state deviations $\mathcal{L}_{\text{data}} = \frac{1}{N_b} \sum_{i=1}^{N_b} \| \hat{\boldsymbol{x}}_i(k+1) - \boldsymbol{x}_i(k+1) \|^2$ and the physics residual loss enforces consistency with the discrete-time approximation of the original ODE dynamics $\mathcal{L}_{\text{physics}} = \frac{1}{N_c} \sum_{j=1}^{N_c} \| \frac{\partial f_{NN}(\tau, \boldsymbol{x_j}(k), \boldsymbol{u_j}(k))}{\partial \tau} - f_{ODE}(\boldsymbol{x}_j(k), \boldsymbol{u}_j(k)) \|^2$, where $f_{ODE}(\boldsymbol{x}_j(k), \boldsymbol{u}_j(k))$ denotes the right-hand side of the mechanistic model in eq. (1).

The weight coefficient is set to $\lambda = 0.1$, which was chosen empirically to balance the magnitudes of the two loss terms such that neither dominates the training process. The physics residual is evaluated at a set of collocation points that coincide with the training data sampling points—specifically, the intermediate time points within each control sampling interval, $t \in \{0.002, 0.005, 0.008, 0.01\}$ h. These collocation points are uniformly distributed across the same input space as the training data, ensuring that the physics constraints are enforced throughout the sampled operating region.

3.3. Multi-step Prediction Implementation

In MPC, it is necessary to predict the state trajectory for the next H sampling periods, where each period has a duration of Δt. Using the PINN, multi-step prediction can be achieved in an iterative manner. Given the current measured state $\boldsymbol{x}_k$ and the future control sequence $\boldsymbol{U}_k = [\boldsymbol{u}_k, \boldsymbol{u}_{k+1}, \cdots, \boldsymbol{u}_{k+H-1}]^T$, the following steps are executed sequentially for $j = 0, \cdots, H-1$:

(1) Construct the input vector: $[\tau = \Delta t, \boldsymbol{x}_{k+j}, \boldsymbol{u}_{k+j}]$ (all need to be normalized).

(2) Run the PINN to obtain $\hat{\boldsymbol{x}}_{k+j+1}$(after inverse standardization).

(3) Use $\hat{\boldsymbol{x}}_{k+j+1}$ as the initial state for the next time step and repeat steps 1–2.

4. Controller Design and Stability Analysis

4.1. Standard MPC Design

4.1.1 Optimization Problem Formulation

At each sampling time t_k, the standard MPC determines the optimal control sequence by solving the following finite-horizon optimization problem:

$$min_{\boldsymbol{u}(k|k),\ldots,\boldsymbol{u}(k+H-1|k)} J(k) = \sum_{j=0}^{H-1}\left[\omega_1\left(\hat{x}_1(k+j+1|k)\right)^2 + \omega_2\left(\hat{x}_2(k+j+1|k)\right)^2\right] \tag{6}$$

where, H is the prediction horizon; $\hat{x}_1(k+j+1|k)$ and $\hat{x}_2(k+j+1|k)$ represent the predicted concentration deviation and temperature deviation at the $(j+1)$-th future prediction sampling step, respectively, forecasted by the PINN; $\omega_1 = 1000$ and $\omega_2 = 1$ are the weighting coefficients; The setpoint is $\boldsymbol{0}$.

Assuming the control horizon equals the prediction horizon ($M = H = 10$), the decision variables consist of the full sequence of control inputs over the next H prediction steps, denoted as

$$\boldsymbol{U}_k = \left[u_1(k|k), u_2(k|k), u_1(k+1|k), u_2(k+1|k), \cdots, u_1(k+H-1|k), u_2(k+H-1|k)\right]^T \in R^{2H} \tag{7}$$

The constraints only include the amplitude limits on the control inputs:

$$\begin{aligned} &-3.5 \le u_1(k+j|k) \le 3.5,\ j = 0,1,\cdots,H-1 \\ &-5\times10^5 \le u_2(k+j|k) \le 5\times10^5,\ j = 0,1,\cdots,H-1 \end{aligned} \tag{8}$$

The predicted states $\hat{\boldsymbol{x}}$ are generated by the PINN model, which takes the current measured state $x(t_k)$ and the control sequence U_k to be optimized as inputs, and outputs the states over the next H sampling periods.

4.1.2 Control Sequence and Implementation

After solving the optimization problem, the optimal control sequence $\boldsymbol{U}_k^*$ is obtained. According to the receding horizon principle of MPC, only the first control action is applied to the actual system:

$$\boldsymbol{u}(t_k) = [u_1^*(k|k), u_2^*(k|k)] \tag{9}$$

Subsequently, within the interval $[t_k, t_k + \Delta t]$, a fine integration step size $h_C = 10^{-4}$h is used to simulate the real CSTR mechanistic model, yielding the next time-step state $x(t_{k+1})$. The above process is repeated until the end of the simulation.

4.2. Design of PINN-DO-LBMPC Controller

4.2.1 Design of PI Disturbance Observer

To eliminate steady-state offsets caused by unmodeled dynamics and disturbances, a PI disturbance observer (PI-DO) is embedded into the MPC prediction loop. The observer equation is given by:

$$\begin{aligned}
&\hat{\boldsymbol{x}}(k \mid k-1) = f_{\mathrm{NN}}(\hat{\boldsymbol{x}}(k-1), \boldsymbol{u}(k-1)) + \hat{\boldsymbol{d}}(k-1) \\
&\boldsymbol{e}(k) = \boldsymbol{x}_{\mathrm{meas}}(k) - \hat{\boldsymbol{x}}(k \mid k-1) \\
&\hat{\boldsymbol{x}}(k) = \hat{\boldsymbol{x}}(k \mid k-1) + \boldsymbol{L}_p \boldsymbol{e}(k) \\
&\hat{\boldsymbol{d}}(k) = \hat{\boldsymbol{d}}(k-1) + \boldsymbol{L}_i \boldsymbol{e}(k)
\end{aligned} \tag{10}$$

where L_p and L_i are gain matrices. The updated prediction model over the horizon $j = 0, \ldots, N-1$ becomes:

$$\hat{\boldsymbol{x}}(k+j+1 \mid k) = f_{\mathrm{NN}}(\hat{\boldsymbol{x}}(k+j \mid k), \boldsymbol{u}(k+j \mid k)) + \hat{\boldsymbol{d}}(k),\ j = 0,1,\ldots,N-1 \tag{11}$$

The observer gain matrices are chosen as diagonal matrices $\mathbf{L}_p = diag(1,1), \mathbf{L}_i = diag(0.35,0.35)$. These values were tuned to balance convergence speed and noise suppression. The convergence of the state estimation error is guaranteed since the eigenvalues of $\mathbf{I} - \mathbf{L}_p$ are zero, which lie within the unit circle. An identical integral gain of 0.35 for both concentration and temperature channels enables balanced and fast disturbance tracking kinetics without triggering unwanted control chatter under measurement noise.

4.2.2 Design of Lyapunov-based MPC

(1) Lyapunov Function and Region of Attraction

To guarantee closed-loop stability, a quadratic Lyapunov function $V(x) = x^T P x = a x_1^2 + d x_2^2 + 2b x_1 x_2$ is defined, with $a = 1060, b = 22, d = 0.52$. The region of attraction is $\Omega_\rho = \{x \in \mathbb{R}^2 \mid V(x) \le \rho\}$. In this paper, $\rho = 2$ is chosen. When the state lies inside Ω_p, the system can be considered to be in a safe region.

(2) Constraint Construction

LBMPC adds the following Lyapunov function-related constraints to standard MPC to ensure that the closed-loop states always contract toward the origin and remain within the region of attraction. Based on the current state position, the constraints are divided into two cases:

Case 1: The current state is inside the region of attraction, i.e. $V\big(x(t_k)\big) < 2$. In this case, it is required that the predicted state $\hat{\boldsymbol{x}}(t_{k+1}|k)$ after one sampling period still satisfies $V(\hat{\boldsymbol{x}}(t_{k+1}|k)) \le 2$ to ensure the state does not leave the region of attraction. Using the final micro-step state $\hat{x}(t_{k+1})$ predicted by the PINN, the constraint is expressed as:

$$V\Big(\boldsymbol{x}(t_{k+1} \mid k)\Big) \le 2 \tag{12}$$

Case 2: The current state is outside the region of attraction, i.e. $V\left(x(t_k)\right) \geq 2$. In this case, it is required that the Lyapunov function value decreases at a certain rate along the closed-loop trajectory to ensure convergence of the state into the region of attraction. The discrete form of the derivative constraint is adopted:

$$\frac{V\left(\boldsymbol{x}(t_{k+1}\mid k)\right)-V(\boldsymbol{x}(t_k))}{\Delta t} \leq -cV(\boldsymbol{x}(t_k)) \tag{13}$$

where, c=15 is the decay coefficient. Rearranging gives:

$$V\left(\boldsymbol{x}(t_{k+1})\right) \leq V(\boldsymbol{x}(t_k))\left(1-c\Delta t\right) \tag{14}$$

These two types of constraints are integrated into the optimization problem through conditional judgment.

(3) Optimization Problem and Solution

At each sampling instant, the LBMPC solves the following optimization problem:

$$\begin{aligned}
&\min_{\boldsymbol{U}_k} J(k) = \sum_{j=0}^{H-1}\left[\omega_1\left(\hat{x}_1(k+j+1\mid k)\right)^2 + \omega_2\left(\hat{x}_2(k+j+1\mid k)\right)^2\right] \\
&\text{s.t.} \quad \boldsymbol{u}(k+j\mid k)\in\mathcal{U}, \quad \forall j=0,\ldots,H-1 \\
&\text{Lyapunov constraints:} \begin{cases} V(\boldsymbol{x}(t_{k+j+1}\mid k)) \leq 2, \forall j=0,\ldots,H-1, & \text{if } V(\boldsymbol{x}(t_k))<2 \\ V(\boldsymbol{x}(t_{k+1}\mid k)) \leq V(\boldsymbol{x}(t_k))(1-c\Delta t), & \text{if } V(\boldsymbol{x}(t_k))\geq 2 \end{cases}
\end{aligned} \tag{15}$$

where, $\hat{x}(t_{k+1}|k)$ is predicted by the PINN model using the current measured state $x(t_k)$ and the candidate control sequence U_k. Since the constraint only involves the state after one sampling period, the code imposes a constraint of $V\left(x\left(t_{k+j+1}\right)\right) \leq 2$ on each prediction step j in case 1, that is, the state of all future sampling points is required to be in the stable domain; Scenario 2 only applies the decline rate constraint to the current step.

The optimization is executed using the Ipopt nonlinear solver, with gradients and Jacobians computed via finite-difference approximation. A warm-start strategy—shifting the optimal solution from the previous time step—is employed to accelerate convergence. Under the standard receding horizon principle, only the first calculated control action is applied to the plant. By incorporating these Lyapunov constraints, the LBMPC mathematically guarantees that the closed-loop states remain within the region of attraction and asymptotically converge to the equilibrium, thereby resolving the potential stability issues inherent in standard MPC.

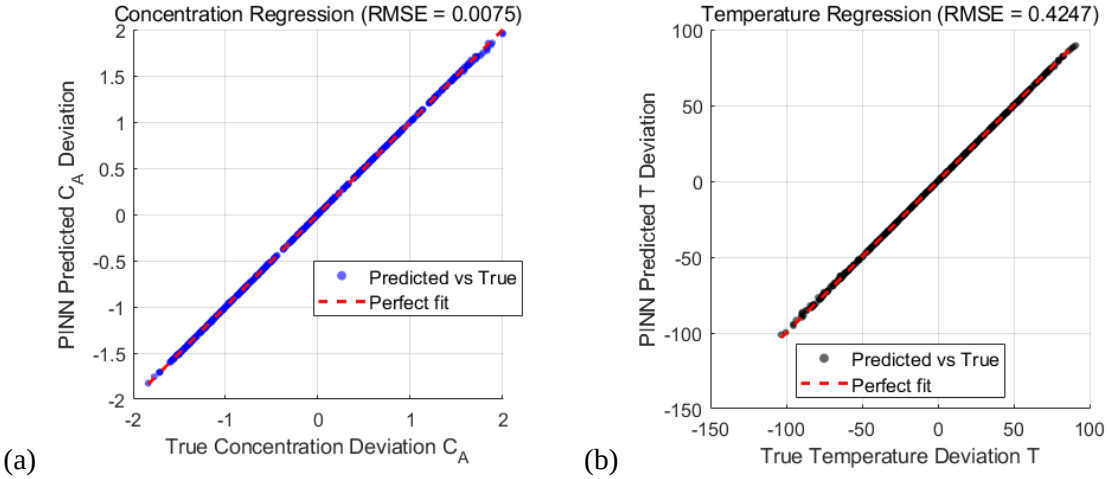
The online optimization problem involves a prediction horizon of H =10. At each sampling instant, the Ipopt solver typically converges within 0.02–0.08 s on a standard desktop computer (Intel Core i7, 16 GB RAM). Since the control sampling period is Δt = 0.01 h = 36 s, the computation time is negligible compared to the sampling interval, confirming that the proposed PINN-DO-LBMPC is suitable for real-time implementation.

5. Simulation Results and Comparative Analysis

5.1. Training and Accuracy Verification of the PINN System Model

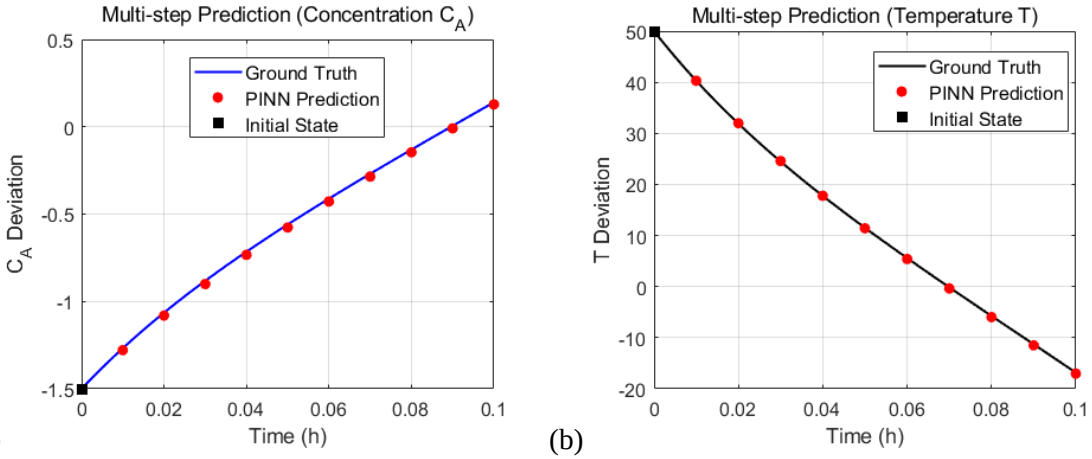
The PINN dynamic model was trained to replace the mechanistic ODEs in the MPC loop. The total loss converged rapidly within few epochs. As shown in Fig. 1 and Fig. 2, the prediction RMSEs

for concentration (x_1) and temperature (x_2) deviations are 0.0075 kmol/m^3 and 0.4247 K, respectively. Multi-step rolling predictions closely align with the ground-truth ODE trajectories across the operating region, confirming that the trained PINN provides sufficient accuracy for predictive control.



(a) (b)

Figure 1. Regression analysis of predicted and true state deviations; (a) concentration deviation regression, and (b) temperature deviation regression

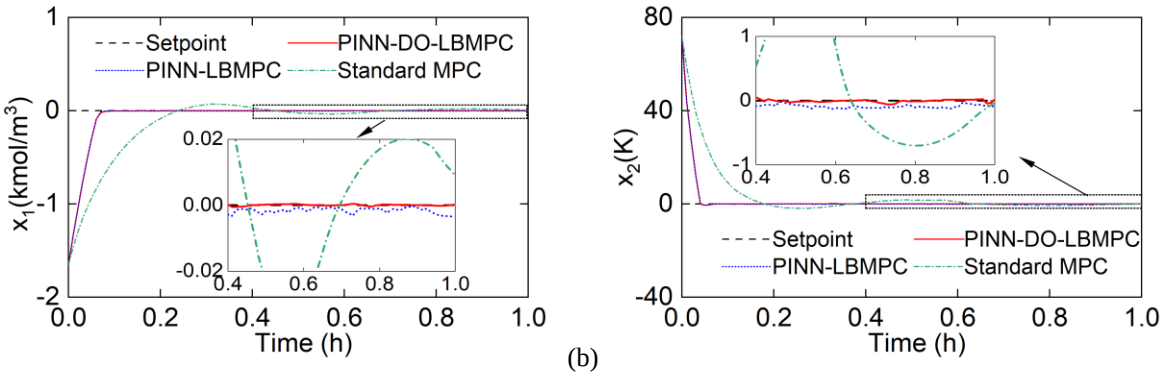


(a) (b)

Figure 2. Multi-step rolling prediction of PINN starting from a non-steady initial state; (a) trajectory of concentration deviation, and (b) trajectory of temperature deviation

5.2. Comparison of Nominal Control Performance under Disturbance-Free Conditions

To evaluate the dynamic control efficacy of the proposed strategy, comparative simulations were conducted among Standard MPC, PINN-LBMPC, and the proposed PINN-DO-LBMPC under nominal, disturbance-free conditions. The tracking performance of Standard MPC, PINN-LBMPC, and proposed PINN-DO-LBMPC was evaluated under nominal, disturbance-free conditions starting far from equilibrium (fig. 3, tab. 2). Standard MPC exhibits slow response with a long settling time ($t_s = 0.17$ h). By embedding Lyapunov constraints, both PINN-LBMPC and PINN-DO-LBMPC achieve fast, smooth convergence to the origin within 0.07 h, reducing settling time by 58.8% with minimal overshoot ($< 1\%$). Overlapping trajectories under nominal conditions confirm that the disturbance observer introduces no unwanted bias under ideal operations.



(a) (b)

Figure 3. Nominal tracking performance of different control schemes; (a) Response of concentration deviations, and (b) Response of temperature deviations

Table 2 Nominal tracking & dynamic performance

Performance Metric	State Variable	Standard MPC	PINN-LBMPC	PINN-DO-LBMPC
Integral Absolute Error	x_1 (kmol/m^3)	0.2148	0.0602	0.0602
	x_2 (K)	4.8210	1.6745	1.5832
Root Mean Square Error	x_1 (kmol/m^3)	0.4125	0.2281	0.2274
	x_2 (K)	15.8201	9.8215	9.7904
Settling Time	t_s (h)	0.17	0.07	0.07
Overshoot	σ%	2.66	0.97	0.88
Steady-State Error	x_1 (kmol/m^3)	0.00903	-0.00347	0.00015
	x_2 (K)	-0.03582	-0.08569	0.01765

5.3. Comparison of Robustness under Unmodeled Disturbances and Measurement Noise

To evaluate system robustness under process uncertainties, a $+10$K step change in feed temperature was introduced at $t = 0.5$ h, along with Gaussian measurement noise ($\sigma = 0.01$ for x_1, $\sigma = 0.2$ for x_2). The closed-loop responses and online disturbance estimations are depicted in fig. 4 and fig. 5, respectively.

As shown in fig. 4, fig. 5, and tab. 3, Standard MPC exhibits severe oscillations and a large steady-state temperature offset (15.76 K). PINN-LBMPC maintains stability via Lyapunov constraints but retains a persistent offset (1.38 K) due to uncompensated dynamics. In contrast, PINN-DO-LBMPC rapidly suppresses the thermal disturbance via feedforward compensation, reducing steady-state temperature error to 0.179 K with minimal overshoot (2.08%) and lowest IAE (1.6120). The PI observer (fig. 5) accurately estimates the thermal disturbance while filtering sensor noise, confirming superior offset-free tracking and robustness.

Table 3 Robustness tracking & dynamic performance

Performance Metric	State Variable	Standard MPC	PINN-LBMPC	PINN-DO-LBMPC
Integral Absolute Error	x_1 (kmol/m^3)	0.3015	0.0625	0.0595
	x_2 (K)	9.8512	2.3345	1.6120
Root Mean Square Error	x_1 (kmol/m^3)	0.4312	0.2285	0.2276
	x_2 (K)	22.4510	9.8820	9.8104
Settling Time	t_s (h)	$>$0.5	0.07	0.07
Overshoot	σ%	25.03	2.44	2.08
Steady-State Error	x_1 (kmol/m^3)	-0.37349	-0.00373	-0.00389
	x_2 (K)	15.76236	1.38228	0.17902

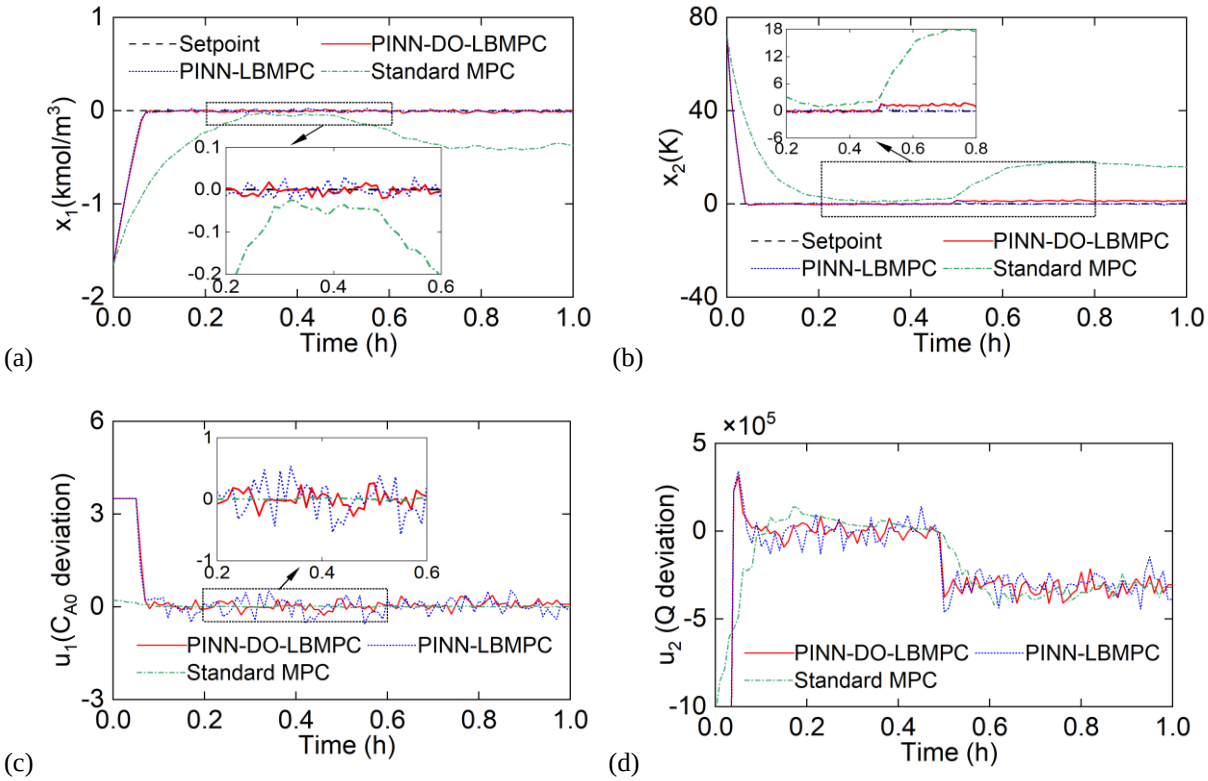


Figure 4. Robust tracking performance of different control schemes; (a) Response of concentration deviations, (b) Response of temperature deviations, (c) Control input u_1, and (d) Control input u_2

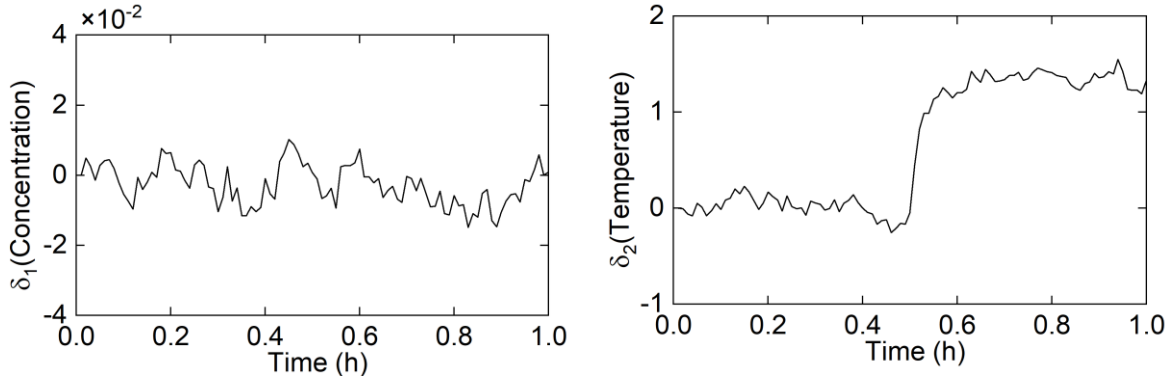


Figure 5. Disturbance estimation of PINN-DO-LBMPC

6. Conclusions

This paper presents a PINN-DO-LBMPC framework for an unstable nonlinear CSTR, integrating physics-informed neural networks, Lyapunov constraints, and a disturbance observer. PINN combines data with physical laws, achieving faster training and higher long-horizon prediction accuracy than purely data-driven models, fully replacing first-principle models as the MPC predictor. A decreasing Lyapunov constraint is explicitly incorporated to keep state trajectories within the stability region, significantly improving convergence speed and smoothness compared to unconstrained MPC. A PI-type disturbance observer estimates unmodeled dynamics in real time; its feedforward compensation, combined with integral action, eliminates steady-state errors under parameter drift and sensor noise, ensuring rapid recovery with minimal fluctuations. In summary, the control strategy proposed in this paper achieves a good balance among the high efficiency of data-driven modeling, the safety guarantee of state convergence, and the disturbance rejection capability under complex industrial environments. It

provides a theoretically rigorous and highly practically valuable solution for the advanced control of nonlinear chemical processes.

References

[1] Ochoa-Tapia, J.A., et al., Effect of the residence time distribution on the dynamical behavior of isothermal continuous stirred tank reactors: A nonlocal modeling approach, *Ind. Eng.Chem. Res.*, 64 (2025), 12, pp. 6433-6444, https://doi.org/10.22201/fq.18708404e.2025.4.90345

[2] Yang, X., Zhou, Y., Optimal tracking neuro-control of continuous stirred tank reactor systems: A dynamic event-driven approach, *IEEE Trans. Artif. Intell.*, 5 (2023), 5, pp. 2117-2126, https://doi.org/10.1109/TAI.2023.3313105

[3] Yang, X., et al., Dynamic decentralized event-triggered tracking control of continuous stirred tank reactor systems, *IEEE Trans. Autom. Sci. Eng.*, 22 (2025), pp. 16632-16643, https://doi.org/10.1109/TASE.2025.3579230

[4] Mukherjee, D., et al., Fractional standalone backstepping control for nonlinear continuous stirred tank reactor, *Int. J. Robust Nonlinear Control*, 35 (2025), 17, pp. 7422-7437, https://doi.org/10.1002/rnc.70058

[5] Deifalla, M.H.H., Gasmelseed, G.A., Genetic algorithm-based PID optimization for ethyl acetate saponification in a continuous stirred tank reactor, *Am. J. Chem. Eng.*, 12 (2024), 6, pp. 123-131, https://doi.org/10.11648/j.ajche.20241206.11

[6] KR, A.G., Mahapatra, S., Design of decentralised PI/PID control algorithm for a benchmark continuously stirred tank reactor system using frequency domain specifications, *Int. J. Model. Identif. Control*, 43 (2023), 4, pp. 336-351, https://doi.org/10.1504/IJMIC.2023.133434

[7] Khalifa, T.R., et al., Interval type-3 fuzzy Wiener model for nonlinear dynamic systems: Application to continuous stirred tank reactor, Chaos Solitons Fractals, 199 (2025), Part 1, pp. 116584, https://doi.org/10.1016/j.chaos.2025.116584

[8] Hapoglu, H., et al., Artificial neural network discrete-time biomass controller for a continuous stirred tank reactor, *J. Comput. Sci.*, (2025), pp. 102688, https://doi.org/10.1016/j.jocs.2025.102688

[9] Zhou, C., et al., Prescribed performance based direct data-driven model predictive control for continuous stirred tank reactor, *J. Process Control*, 157 (2026), pp. 103609, https://doi.org/10.1016/j.jprocont.2025.103609

[10] Chi, M., et al., The HVAC model predictive control of building thermal dynamic characteristics, Thermal Science, 27 (2023), 2B, pp. 1007-1014, https://doi.org/10.2298/TSCI2302007C

[11] Zrigan, A.F., Control of CSTR using active disturbance rejection control and model predictive control, *Proceedings*, 2024 IEEE 4th International Maghreb Meeting of the Conference on Sciences and Techniques of Automatic Control and Computer Engineering (MI-STA), 2024, pp. 182-187, https://doi.org/10.1109/MI-STA61267.2024.10599687

[12] Subramanian, S., et al., Robust tube-enhanced multi-stage NMPC with stability guarantees, *IEEE Control Syst. Lett.*, 6 (2021), pp. 1112-1117, https://doi.org/10.1109/LCSYS.2021.308950

2

[13] Boufrioua, H., et al., Robust tube-based economic model predictive control of nonlinear syste ms using linear parameter varying approach, *J. Franklin Inst.*, 363 (2026), 1, pp. 108409, http s://doi.org/10.1016/j.jfranklin.2026.108409

[14] Hong, B., et al., Observer-based disturbance rejection tube model predictive control for nonli near systems, *Nonlinear Dyn.*, 113 (2025), 20, pp. 27691-27704, https://doi.org/10.1007/s1107 1-025-11521-5

[15] Gruber, J.K., et al., Min–Max MPC based on an upper bound of the worst case cost with gu aranteed stability. Application to a pilot plant, J. Process Control, 21 (2011), 1, pp. 194-204, https://doi.org/10.1016/j.jprocont.2010.11.016

[16] Abdelsalam, Y., et al., A simplified implementation of tube-enhanced multi-stage NMPC, *IF AC-PapersOnLine*, 53 (2020), 2, pp. 11563-11569, https://doi.org/10.1016/j.ifacol.2020.12.634

[17] Xu, B., et al., Stability-guaranteed model predictive control for UAVs: A survey of the Lyap unov-based framework, Chin. J. Aeronaut., 39 (2026), 1, pp. 104117, https://doi.org/10.1016/j. cja.2026.104117

[18] Mitić, V.V., et al., Sintering temperature influence on grains function distribution by neural network application, *Thermal Science*, 26 (2022), 1, pp. 299-307, https://doi.org/10.2298/tsci21 0420283m

[19] Hassanpour, H., et al., Artificial neural network-based model predictive control using correlat ed data, *Ind. Eng. Chem. Res.*, 61 (2022), 8, pp. 3075-3090, https://doi.org/10.1021/acs.iecr.1c 04339

[20] Xu, C., Data-driven model predictive control based on LSTM and PSO for the continuous st irred tank reactor system, *Eng. Res. Express*, 8 (2026), 12, pp. 125004, https://doi.org/10.1088 /2631-8695/ae7aee

[21] Velioglu, M., et al., Physics-informed neural networks for dynamic process operations with li mited physical knowledge and data, *Comput. Chem. Eng.*, 192 (2025), pp. 108899, https://doi. org/10.1016/j.compchemeng.2024.108899

[22] Luo, K., et al., Physics-informed neural networks for PDE problems: A comprehensive revie w, *Artif. Intell. Rev.*, 58 (2025), 10, pp. 323, https://doi.org/10.1007/s10462-025-11322-7

[23] Lei, X.H., He, J.H., Frontiers in thermal science driven by artificial intelligence, *Thermal Sci ence*, 29 (2025), 3, pp. 1671-1677, https://doi.org/10.2298/TSCI250101059L

[24] Anbu, S., et al., Design of a multiloop controller for a nonlinear process, *Int. J. Adv. Compu t. Sci. Appl.*, 13 (2022), 4, https://doi.org/10.14569/IJACSA.2022.0130434

[25] Wu, Z., et al. Machine-learning-based predictive control of nonlinear processes. Part II: Com putational implementation. *AIChE Journal*, 65(2019) , pp.e16734, https://doi.org/10.1002/aic.1 6734

RECEIVED DATE: 16.5.2026

DATE OF CORRECTED PAPER: 01.7.2026.
DATE OF ACCEPTED PAPER: 10.9.2026.