Journal of Process Control 143 (2024) 103315

Contents lists available at ScienceDirect

Journal of Process Control

journal homepage: www.elsevier.com/locate/jprocont

Physics-informed neural networks for multi-stage Koopman modeling of microbial fermentation processes

Quan Li, Jingran Zhang, Haiying Wan, Zhonggai Zhao, Fei Liu *

Key Laboratory of Advanced Process Control for Light Industry (Ministry of Education), Institute of Automation Jiangnan University, Wuxi, 214122, PR China

ARTICLE INFO

Keywords:
Microbial fermentation
Growth stages
Multi-stage Koopman modeling
Fuzzy C-means clustering
Physics-informed neural networks

ABSTRACT

This paper investigates the modeling problem of microbial fermentation suitable for model-based control design techniques. Given the evident nonlinear and stage characteristics of microbial fermentation processes, a single data-driven model cannot fully capture microbial growth characteristics. Therefore, we propose a multi-stage Koopman modeling method based on physics-informed neural networks. Initially, the fuzzy C-means clustering algorithm is employed to partition the microbial growth stages. Subsequently, the Koopman operator is approximated through physics-informed neural networks. Utilizing the Koopman operator to map the dynamic behavior of the microbial fermentation system into a high-dimensional linear space, and modeling each growth stage separately in the linear space. Compared to conventional neural network methods, physics-informed neural networks integrate the advantages of physical models and neural networks, thereby better preserving the dynamic information of microbial growth and enhancing the model's generalization performance. A penicillin fermentation case study verifies the effectiveness of our proposed method.

1. Introduction

Microbial fermentation, as an important biotransformation technology, has broad application prospects and significant economic value in industrial production and biopharmaceutical fields [1]. Through the metabolic activities of microbial strains under suitable environmental conditions, various useful compounds such as biofuels, pharmaceuticals, enzymes, and organic acids can be produced. However, the complexity of microbial metabolic networks, the variability of environmental conditions, and the various biochemical reactions during fermentation make it difficult to ensure consistent product quality and yield in actual production. To optimize and control the quality and yield of fermentation products, it is necessary to develop a model that can accurately capture the dynamic characteristics of the system to describe the process [2]. This need poses significant challenges for modeling fermentation processes.

Many scholars study the mechanistic characteristics of the fermentation process and model it from the perspective of microbial metabolism. The kinetic model of the fermentation process is constructed based on the Monod model [3] and the Michaelis–Menten equation [4], which describe the relationship between cell-specific growth rate and limiting substrate concentration. Bajpai and Reuss [5] established a substrate inhibition kinetic model for penicillin fermentation based on the Monod model, considering substrate consumption, biomass growth, and product formation. Building on this, Goldrick et al. [6] introduced Raman spectroscopy equipment and considered the effects of nitrogen and phenylacetic acid concentrations on biomass and penicillin yield, developing an industrial-scale simulator (IndPenSim). de Andrés-Toro et al. [7] developed a mechanistic model for beer fermentation based on the Monod model. Through isothermal experiments at five different temperatures, they demonstrated that the kinetic model could accurately fit the experimental data and predict non-isothermal experimental data with high precision. Sakimoto et al. [8] used the Langmuir adsorption equation and the Michaelis–Menten equation to establish a kinetic model for ethanol fermentation. The model can simultaneously predict the theoretical curves of ethanol fermentation and saccharification hydrolysis. Despite the achievements of mechanistic modeling in capturing the complex dynamic behavior of fermentation processes, constructing such models requires a detailed understanding of the underlying mechanisms. The highly nonlinear and complex nature of microbial fermentation processes makes it challenging to develop models that can fully describe mechanisms.

In recent years, with advancements in data acquisition and processing technologies, data-driven modeling has gradually become an important approach to addressing the challenges of fermentation process modeling by directly learning the dynamic characteristics of the system from experimental data [9]. However, while data-driven modeling can quickly learn the behavioral characteristics of a system, it often

* Corresponding author.
E-mail address: fliu@jiangnan.edu.cn (F. Liu).

https://doi.org/10.1016/j.jprocont.2024.103315
Received 31 May 2024; Received in revised form 25 August 2024; Accepted 8 September 2024
Available online 19 September 2024

lacks a deep understanding of the underlying structure and evolution patterns of the system. Moreover, black-box models established through data-driven approaches are typically not suitable for model-based control and optimization studies [10]. Koopman modeling helps reveal the fundamental dynamical characteristics of a system, providing more interpretable models oriented towards control [11,12]. However, manual selection of the Koopman operator requires a deep understanding and expertise in the system [13]. Due to the complexity and nonlinear nature of microbial fermentation processes, manually selected operators may not fully capture the system's dynamic behavior, thus affecting the accuracy and reliability of the model. The papers [14,15] combined the flexibility of deep neural network (DNN) architectures with the structural advantages of Koopman theory, learning Koopman feature functions from data through deep learning. However, while the introduction of DNN addresses the issue of Koopman operator selection, the operators obtained through network approximation may cause the model to lose its dynamical characteristics, reducing the model's robustness in predicting processes under various operating conditions. Combining prior knowledge with data-driven models through hybrid modeling approaches can effectively enhance the performance of data-driven models [16,17]. Recent researches have proposed integrating known physical information into machine learning and deep learning techniques to develop a hybrid physically-driven dynamic modeling framework. The framework leverages the advantages of both physical and data-driven modeling for online monitoring, prediction, and optimization of biological processes [18,19]. Physics-Informed Neural Networks (PINN) adopts the framework by combining mechanistic equations with DNN to improve the dynamical characteristics of the model, integrating the accuracy of mechanistic modeling with the flexibility of data-driven modeling, thus enhancing the model's prediction performance and applicability [20–22]. PINN+Koopman In the field of biological process modeling, PINN is employed to simulate phenomena such as mass transfer and reaction kinetics in cell culture processes. Bangi et al. [23] incorporated known derivatives of fermentation process kinetics into the network to establish a hybrid model for batch production of β-carotene using Saccharomyces cerevisiae. Cui et al. [24] acquired the dynamic evolution model of Chinese hamster ovary (CHO) cell bioreactors from process data by integrating physical laws (such as mass balance) with dynamic expressions of metabolic fluxes. However, the aforementioned studies mentioned above solely focused on utilizing physical information and did not account for the applicability of the model to model-based controllers. We employ PINN to select the Koopman operator. Subsequently, a microbial fermentation process model suitable for model-based controllers is established in high-dimensional space.

Using the Koopman method to establish a model for microbial fermentation processes presents another challenge due to its apparent stage characteristics. Even methods based on PINN find it difficult to establish a global linear model to accurately describe the dynamic characteristics of all stages. To better describe the process characteristics of each stage, it is necessary to model each growth stage separately [25,26]. In cases where the microbial growth stages are unknown, data clustering analysis methods can be used to partition the stages based on historical process data [27]. However, due to the highly nonlinear nature of fermentation processes, the boundaries between clusters used by typical clustering algorithms may not be clearly defined [28]. The Fuzzy C-Means (FCM) algorithm provides a better and more helpful method for partitioning growth stages in fermentation processes [29]. Alfredo and Marco [30] integrated fuzzy modeling and the FCM algorithm to establish a fuzzy model for the non-structured non-segregated bioreactor process of ethanol production from glucose. [31] used FCM to partition various stages of baker's yeast fermentation process, effectively addressing the issue that traditional control methods cannot meet the practical needs of fermentation process control. Therefore, in-depth research on each growth stage of fermentation processes can significantly improve modeling accuracy and control effectiveness. However, previous studies have only focused on modeling each stage, lacking guidance from physical information and considering the model's applicability for model-based controllers. We combine FCM with multi-stage Koopman modeling based on PINN to establish the microbial fermentation process model.

This paper proposes a multi-stage Koopman modeling method based on PINN by incorporating physical information and microbial fermentation stage characteristics. Firstly, the FCM algorithm is employed to analyze historical data of the microbial fermentation process to divide each growth stage. Subsequently, the known dynamics information of the fermentation process is introduced into the DNN to construct the PINN. Furthermore, the multi-stage Koopman modeling method based on PINN is separately applied to construct stage models for each growth stage. Finally, a penicillin case study is used to verify the effectiveness of the proposed modeling approach. The novelty of this study can be summarized as follows:

(1) Establishing independent Koopman models for each growth stage ensures that the model accurately captures the dynamics of microbial growth.
(2) Introducing physical information into neural networks allows the model to maintain high-precision predictions under different operating conditions, enhancing the model's robustness and applicability.
(3) Using PINN to approximate the Koopman operator for the first time enhances the model's physical consistency and predictive performance.

The remainder of this paper is organized as follows. The algorithm for dividing microbial fermentation growth stages is described in Section 2. Section 3 introduces the PINN-based multi-stage Koopman modeling method. Section 4 presents detailed simulation results for the penicillin case study. Finally, Section 5 summarizes this paper.

2. Microbial fermentation process growth stage division

The microbial fermentation process goes through multiple stages, each with unique dynamic characteristics that are crucial for the accuracy and predictive performance of the model. To capture the stage characteristics accurately, Section 2.1 will detail the characteristics of the growth stages and analyze the dynamic behavior. Subsequently, Section 2.2 will describe in detail how the FCM algorithm is applied to divide the fermentation process into different stages.

2.1. Growth stage characteristics

The microbial fermentation process involves a series of biochemical reactions facilitated by specific strains of microorganisms in active microbial cells. Typically, the bacteria usually need to go through the growth adaptation phase, logarithmic growth phase, growth stability phase and cell death phase. Fig. 1 shows the changes in the four stages of bacterial concentration. During the growth adaptation phase, microorganisms adapt to their environment, with relatively low growth and metabolic rates. In the logarithmic growth phase, microorganisms divide and metabolize rapidly at their maximum rate. In the growth stability phase, the depletion of nutrients causes growth to halt and metabolic activity to decrease. Finally, during the cell death phase, a significant number of microorganisms die, and metabolic activity sharply declines.

The dynamic behaviors of different stages in the microbial fermentation process vary significantly. Using a single model may lead to inaccurate descriptions of certain stage characteristics, thereby affecting the overall model's accuracy and predictive performance. Multi-stage models can be employed to establish more adaptive sub-models tailored to the characteristics of different stages. By modeling each stage separately, it is possible to capture the dynamic characteristics of each stage more accurately, thereby improving the overall model's

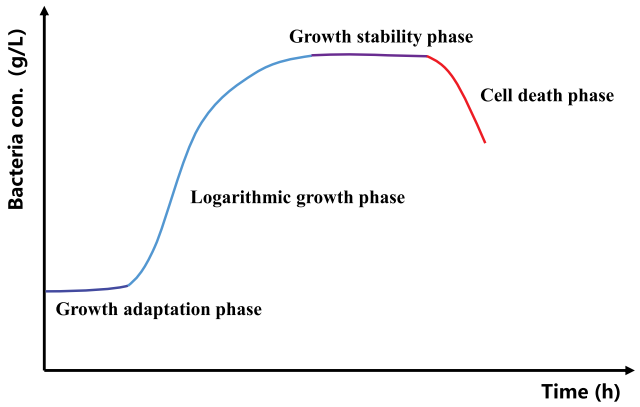


Fig. 1. Microbial growth cycle curve.

predictive performance and control precision. For example, during the logarithmic growth phase, the model can focus on rapid growth characteristics, while during the stable and death phases, it can better capture equilibrium states and decline processes.

2.2. Growth stage division

In the microbial fermentation process, microorganisms produce different metabolites at various growth stages, and the concentration changes of metabolites can reflect the characteristics of microorganisms at different stages. Although the concentration variables of substrates, biomass, and products in microbial fermentation can well reflect the stage characteristics of the fermentation process, manually dividing the fermentation process into different stages based on these variables requires extensive experience and cannot be verified. Therefore, this paper aims to use the FCM clustering algorithm to cluster sampled data and extract stage information from concentration variables to divide the microbial fermentation process.

The Fuzzy C-Means clustering algorithm is a clustering method based on function optimization. It optimizes a fuzzy objective function to determine the membership degree of each sample point to the cluster centers, thereby determining the affiliation of the sample points [29]. This paper sets the cluster center c to the growth stage number r. Membership matrix $U = \{u_{tj}\}, (t = 1, 2, \dots, n, j = 1, 2, \dots, r)$, u_{tj} ranges between $[0, 1]$, and the total sum of membership for each data point satisfies:

$$\sum_{t=1}^{n}\sum_{j=1}^{r} u_{tj} = 1. \tag{1}$$

Calculating the membership values $\{u_{tj}\}^k$ for the kth iteration:

$$\{u_{tj}\}^k = \begin{cases} \dfrac{1}{\sum_{l=1}^{r}\left(\frac{\{d_{tj}\}^k}{\{d_{tl}\}^k}\right)^{2/(m-1)}} & , \{d_{tj}\}^k \neq 0 \\ 1 & , \{d_{tj}\}^k = 0 \end{cases}, \ t = 1, 2, \dots, n, \ j = 1, 2, \dots, r, \tag{2}$$

where $\{d_{tj}\}^k$ represents the Euclidean distance between data $x_t (t = 1, 2, \dots, n)$ and cluster centers $\{c_j\}^k (j = 1, 2, \dots, r)$, m is the fuzzy exponent ($m > 1$, and here m is taken as 2).

Updating the cluster centers $\{c_j\}^k$ based on $\{u_{tj}\}^k$:

$$\{c_j\}^k = \frac{\sum_{t=1}^{n}\{u_{tj}^m\}^k g_i}{\sum_{t=1}^{n}\{u_{tj}^m\}^k}, \ j = 1, 2, \dots, r. \tag{3}$$

Set a criterion stopping threshold ϑ to determine whether the cluster centers $\{c_j\}^k$ have converged to a stable value. If the difference between the current iteration's cluster centers and the previous iteration's cluster centers $|\{c_j\}^k - c_j^*| \leq \vartheta$, then the change in the cluster centers is considered negligible. The algorithm stops updating and outputs $\{c_j\}^k$ and u_{tj}^k. Otherwise, update $c_j^* = \{c_j\}^k$, set $k = k + 1$, and perform iterations based on Eqs. (2) and (3) until the iteration stopping criterion is met.

3. PINN-based multi-stage Koopman modeling

The unknown nonlinear dynamics of microbial fermentation process are given by the following equation:

$$\frac{dx(t)}{dt} = f(x(t), u(t)), \tag{4}$$

where $x(t) \in R^{n_x}$ is the object state vector, $u(t) \in R^{n_u}$ is the control input vector, n_x and n_u are the numbers of states and control inputs respectively. f is a set of unknown differential equations representing the correlations between process states and control inputs.

The Eq. (4) describes a nonlinear system in continuous time, which captures the complex nonlinear biochemical process of microbial fermentation. However, in practical industrial processes, data collection and control operations are typically conducted at discrete time intervals. To facilitate numerical simulation and implementation of the model on a computer, it is necessary to convert the continuous-time system in Eq. (4) into a discrete-time form to represent the microbial fermentation system. The process of discretizing the system (4) at discrete time points $t_k = k \cdot \Delta t$ using the Euler method with a time step Δt is as follows:

$$x(t_k + \Delta t) = x(t_k) + \Delta t \cdot f(x(t_k), u(t_k)) = f_d(x(t_k), u(t_k)), \tag{5}$$

where f_d is an unknown nonlinear mapping.

In this section, we will explore how to integrate PINN with multi-stage Koopman modeling to enhance the modeling and prediction of microbial fermentation processes. Multi-stage Koopman modeling captures the system's complex dynamic features by constructing sub-models for different stages. PINN introduces physical laws into neural networks, effectively retaining the system's dynamic characteristics. Section 3.1 will cover the principles and applications of the multi-stage Koopman model. Subsequently, Section 3.2 will detail how PINN is incorporated into the multi-stage Koopman model, discussing how physical information enhances the model's accuracy and stability.

3.1. Multi-stage Koopman model

In the infinite-dimensional Hilbert space $\mathcal{F}$, nonlinear systems can be equivalently represented as linear systems through the Koopman operator $\mathcal{K} : \mathcal{F} \to \mathcal{F}$ [32]. $\mathcal{F}$ is composed of real-valued observable functions $\phi : \mathbb{R}^x \to \mathbb{R}$. Considering the stage characteristics of the microbial fermentation process, we transform the nonlinear evolution of the state $x(t_k)$ determined by f_d in stage $(j = 1, 2, \dots, r)$ into a linear evolution of the observable function ϕ_j determined by $\mathcal{K}_j$:

$$(\mathcal{K}_j\phi_j)(x(t_k), u(t_k)) = \phi_j(f_d(x(t_k), u(t_k)), u(t_k + \Delta t)), (t_k, t_k + \Delta t) \in c_j, j = 1, 2, \dots, r. \tag{6}$$

In microbial fermentation processes, the input $u(t_k)$ is typically coupled with the system state $x(t_k)$. Therefore, we consider the input as part of the system state, thereby extending the state vector of system (5) to include both input and state variables $\chi(t_k) = [x(t_k), u(t_k)]^T$. System (5) can be represented as:

$$\chi(t_k + \Delta t) = F(\chi(t_k)) := \begin{bmatrix} f_d(x(t_k), u(t_k)) \\ \mathcal{S}u(t_k) \end{bmatrix}, \tag{7}$$

where $\mathcal{S}$ is a shift operator, satisfying $u(t_k + \Delta t) = \mathcal{S}u(t_k)$. Eq. (6) can be expressed as:

$$(\kappa_j\phi_j)(\chi(t_k)) = \phi(\chi(t_k + \Delta t)), \ (t_k, t_k + \Delta t) \in c_j, j = 1, 2, \dots, r. \tag{8}$$

Since the Koopman operator is an infinite-dimensional linear operator, analyzing and controlling systems in an infinite-dimensional space lack practical applicability. To investigate the true "state variables" of

the system, it is necessary to find an invariant subspace of the infinite-dimensional function space that includes all system state variables. Within this invariant subspace, the dynamic performance of the system can be studied. Define the subspace $\mathcal{F}$ spanned by $n_\psi > n_x$ linearly independent basis functions $\{\psi_i : \mathbb{R}^{n_x} \to \mathbb{R}\}_1^{n_\psi}$ in $\mathcal{F}$. For $\forall \bar{\phi} \in \overline{\mathcal{F}}$, express $\bar{\phi}$ as a linear combination of these basis functions:

$$\bar{\phi}_j = \sum_{i=1}^{n_\psi} \lambda_{i,j} \psi_{i,j}, \tag{9}$$

where $\lambda_j = \left[\lambda_{1,j}, \lambda_{2,j}, \ldots, \lambda_{n_\psi,j}\right]^T$ is a set of weight coefficients. Let $\Psi_j = \left[\psi_{1,j}, \psi_{2,j}, \ldots, \psi_{n_\psi,j}\right]^T$, then $\bar{\phi}$ is expressed as:

$$\bar{\phi}_j = \Psi_j^T \lambda_j. \tag{10}$$

For $\forall \tilde{\phi} \in \tilde{\mathcal{F}} \subset \mathcal{F}$, the finite-dimensional approximation of the Koopman operator satisfies:

$$\left(K_j \tilde{\phi}_j\right)\left(\chi\left(t_k\right)\right) = \tilde{\phi}_j\left(\chi\left(t_k + \Delta t\right)\right), \quad \left(t_k, t_k + \Delta t\right) \in c_j, j = 1, 2, \ldots, r, \tag{11}$$

where $K_j \in \mathbb{R}^{n_\psi \times n_\psi}$ is the Koopman operator acting on $\mathcal{F}$ in stage j ($j = 1, 2, \ldots, r$).

Substituting Eq. (10), we get:

$$K_j \lambda_j^T \Psi_j(\chi\left(t_k\right)) = \lambda_j^T \Psi_j(\chi\left(t_k + \Delta t\right)). \tag{12}$$

Then,

$$K_j \Psi_j(\chi\left(t_k\right)) = \Psi_j(\chi\left(t_k + \Delta t\right)), \tag{13}$$

$$\Psi_j(\chi\left(t_k\right)) = \begin{bmatrix} x\left(t_k\right) \\ \varphi_j\left(\chi\left(t_k\right)\right) \\ u\left(t_k\right) \end{bmatrix}, \tag{14}$$

where $\Psi_j(\chi\left(t_k\right))$ is the observable in high-dimensional space, and $\varphi_j(\chi\left(t_k\right)) = [\psi_{n_x+1,j}(\chi\left(t_k\right)), \psi_{n_x+2,j}(\chi\left(t_k\right)), \ldots, \psi_{n_x+n_\varphi,j}(\chi\left(t_k\right))]^\top$ is the observed new basis function. n_φ is the number of variables in the high-dimensional space.

Suppose the duration of each stage in a microbial fermentation batch is $f_j (j = 1, 2, \ldots, r)$, $t_0 = 0 \cdot \Delta t$ is the initial time, and each stage has $\frac{f_j}{\Delta t}$ sets of data. Observing data for all states, we obtain:

$$\Psi_{a,j} := \left[\Psi_j\left(\chi\left(t_0\right)\right), \Psi_j\left(\chi\left(t_1\right)\right), \ldots, \Psi_j\left(\chi\left(t_{(f_j/\Delta t)-1}\right)\right)\right], \tag{15}$$

$$\Psi_{b,j} := \left[\Psi_j\left(\chi\left(t_1\right)\right), \Psi_j\left(\chi\left(t_2\right)\right), \ldots, \Psi_j\left(\chi\left(t_{f_j/\Delta t}\right)\right)\right]. \tag{16}$$

According to Eq. (13), we have:

$$K_j \Psi_{a,j} = \Psi_{b,j}. \tag{17}$$

Then,

$$K_j = \Psi_{b,j} {\Psi_{a,j}}^\dagger, \tag{18}$$

where ${\Psi_{a,j}}^\dagger$ is the Moore–Penrose generalized inverse. Therefore, the finite-dimensional approximation of the Koopman operator can be represented as:

$$K_j = \begin{bmatrix} A_{\phi,j} & B_{\phi,j} \\ K_{21,j} & K_{22,j} \end{bmatrix}, \ j = 1, 2, \ldots, r, \tag{19}$$

where $A_{\phi,j} \in \mathbb{R}^{(n_x+n_\varphi)\times(n_x+n_\varphi)}$, $B_{\phi,j} \in \mathbb{R}^{(n_x+n_\varphi)\times n_u}$, $K_{21,j} \in \mathbb{R}^{n_u\times(n_x+n_\varphi)}$, $K_{22,j} \in \mathbb{R}^{n_u\times n_u}$. Let $z\left(t_k\right) = \left[x\left(t_k\right)^T, \varphi_j\left(\chi\left(t_k\right)\right)^T\right]^T \in \mathbb{R}^{n_x+n_\varphi}$, the linear Koopman model representation of the controlled microbial fermentation process is given by:

$$z\left(t_k + \Delta t\right) = A_{\phi,j} z\left(t_k\right) + B_{\phi,j} u\left(t_k\right), \quad \left(t_k, t_k + \Delta t\right) \in c_j, j = 1, 2, \ldots, r, \tag{20}$$

$$\hat{\chi}\left(t_k + \Delta t\right) = C_j \varphi_j\left(\chi\left(t_k + \Delta t\right)\right), \tag{21}$$

where $C_j \in \mathbb{R}^{(n_x+n_\varphi)\times n_x}$ is the mapping matrix from the high-dimensional space to the low-dimensional space, and $\hat{\chi}\left(t_k + \Delta t\right) \in \mathbb{R}^{n_x+n_\varphi}$ is the estimated value of the original system state and control input at the next time.

Combining Eqs. (20) and (21), we can construct the microbial fermentation system (5). However, traditional Koopman modeling methods require manual selection of basis functions (9), which is subjective and may not accurately capture the nonlinear characteristics of the system. Therefore, we will use PINN to learn the basis functions of system (8) and the corresponding Koopman operator.

3.2. Physics-informed neural networks

By incorporating physical information into the neural network to approximate the Koopman operator, the model not only improves physical consistency but also enhances its generalization capability and robustness in complex nonlinear systems. This ensures that the model can accurately capture the dynamic behavior of the system while adhering to physical laws. A schematic diagram of the PINN-based deep Koopman network structure, as shown in Fig. 2, is designed. The network $\theta_{1,j}$ is used to learn the basis functions $\varphi_j\left(\chi\left(t_k\right)\right)$, which combine the original space state $x(t_k)$ and control input $u(t_k)$ at the current time t_k to construct the observation function $\bar{\phi}_j$:

$$\bar{\phi}_j\left(\chi\left(t_k\right)\right) = \left[z\left(t_k\right)^T, u\left(t_k\right)^T\right]^T, \ t_k \in c_j, j = 1, 2, \ldots, r, \tag{22}$$

$$z\left(t_k\right) = \left[x\left(t_k\right)^T, \theta_{1,j}\left(\chi\left(t_k\right)\right)^T\right]^T. \tag{23}$$

To learn the Koopman operator K, a hidden layer without activation functions and biases is constructed. Furthermore, after computing $z(t_k + \Delta t)$ according to Eq. (20), the network $\theta_{2,j}$ is used to reconstruct the original space state $x(t_k + \Delta t)$ and control input $u(t_k + \Delta t)$, as shown in Eq. (21).

By training deep neural networks using historical process data, we can obtain θ_1, θ_2 and K_j, thereby constructing a physically meaningful multi-stage Koopman model based on the physical information neural network, as shown in Eq. (20). To ensure the accuracy of the model, we define the loss function of the network during training as:

$$Loss_j = \lambda_{D,j} Loss_{D,j} + \lambda_{P,j} Loss_{P,j}, \ j = 1, 2, \ldots, r, \tag{24}$$

where $Loss_{D,j}$ and $Loss_{P,j}$ represent the data matching loss and the physical consistency loss, respectively. The $Loss_{D,j}$ ensures high accuracy of the model on observed data, while the $Loss_{P,j}$ incorporates physical constraints of the system, enhancing the model's adherence to physical laws. The combination improves the model's accuracy, robustness, and generalization ability, ensuring that the model not only predicts data accurately but also maintains physical plausibility.

The construction of the data matching loss function $Loss_D$ ensures that the model can reconstruct the original system's state and inputs while maintaining the system's linear behavior in the Koopman space and accurately predicting future states. Therefore, for $\forall \left(t_k, t_k + \Delta t\right) \in c_j, j = 1, 2, \ldots, r$, we formulate three types of cost functions:

1. After mapping the original space with network $\theta_{1,j}$, to ensure accuracy under the high-dimensional linear model composed of the Koopman operator K_j, it is necessary to restrict the prediction error of the state and control inputs at future times in the high-dimensional space. We introduce the linear mapping constraint loss function:

$$Loss_{re,j} = \left\| \begin{array}{l} K_j\left[x\left(t_k\right)^T, \theta_{1,j}\left(\chi\left(t_k\right)\right)^T, u\left(t_k\right)^T\right]^T - \\ \left[x\left(t_k + \Delta t\right)^T, \theta_{1,j}\left(\chi\left(t_k + \Delta t\right)\right)^T, u\left(t_k + \Delta t\right)^T\right]^T \end{array} \right\|^2. \tag{25}$$

2. At the current time t, to ensure that there is not too much information loss under the mapping of the DNN, it is necessary to restrict the reconstruction error of the original space state variables and control inputs. We introduce the state and input reconstruction error loss function:

$$Loss_{li,j} = \left\| \theta_{2,j}\left(\theta_{1,j}\left(\chi\left(t_k\right)\right)\right) - \chi\left(t_k\right) \right\|^2. \tag{26}$$

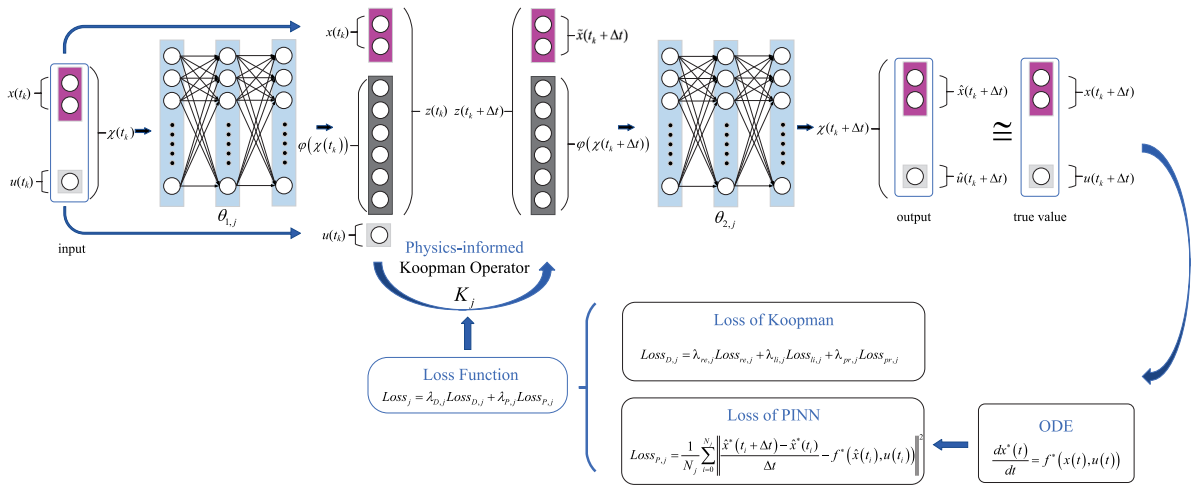


Fig. 2. Structure diagram of the basis functions and corresponding Koopman operator learned by PINN for the system (5).

3. In future state prediction, to enhance the stability of learning by the deep neural network, it is essential to restrict the reconstruction error of the original space state variables and control inputs. We introduce the future state prediction error loss function:

$$Loss_{pr,j} = \left\| \theta_{2,j} \left(K_j \left[x(t_k)^T, \theta_{1,j}(\chi(t_k))^T, u(t_k)^T \right]^T \Big|_{n_x+1}^{n_x+n_\varphi} \right) - \chi(t_k + \Delta t) \right\|^T, \tag{27}$$

where $\bullet|_{n_x+1}^{n_x+n_\varphi}$ represents the column elements of the vector from the (n_x+1)-th to the (n_x+n_φ)-th columns.

Combining the three data-related loss functions, $Loss_{D,j}$ can be expressed as:

$$Loss_{D,j} = \lambda_{re,j} Loss_{re,j} + \lambda_{li,j} Loss_{li,j} + \lambda_{pr,j} Loss_{pr,j}, \tag{28}$$

where $\lambda_{re,j}$, $\lambda_{li,j}$ and $\lambda_{pr,j}$ are weight coefficients used to balance the contributions of each component loss to the total loss.

The construction of the physics consistency loss function $Loss_P$ aims to ensure that the model accurately reflects the physical behavior of the fermentation process, thereby improving the modeling accuracy and generalization ability of the model for the fermentation process. Therefore, we need to devise a physics consistency loss function that ensures the model's dynamic consistency with the actual fermentation process. Assuming that the physical information of the fermentation process is described by the following form of ordinary differential equations:

$$\frac{dx^*(t)}{dt} = f^*(x(t), u(t)). \tag{29}$$

The physics consistency loss function $Loss_{P,j}$ can be defined as the difference between the predicted derivatives by the model and the dynamics equations:

$$Loss_{P,j} = \frac{1}{N_j} \sum_{i=0}^{N_j} \left\| \frac{\hat{x}^*(t_i + \Delta t) - \hat{x}^*(t_i)}{\Delta t} - f^*(\hat{x}(t_i), u(t_i)) \right\|^2,$$
$$t \in c_j, j = 1, 2, \ldots, r, \tag{30}$$

where $N_j = \frac{f_j}{\Delta t}$ is the number of samples, and t_i represents the time step.

4. Penicillin fermentation case study

The proposed multi-stage Koopman modeling method based on PINN is applied to the penicillin production process, aiming to accurately predict the dynamic behavior of the fermentation process. To generate experimental data for establishing the multi-stage Koopman model, this paper utilizes a process simulator described by Bajpai et al. based on substrate inhibition kinetics [3,33]. The simulator is represented by the following set of ordinary differential equations:

Table 1
Initial operating conditions for the process simulator.

Biomass concentration (X_0)	0.1 (g/L)
Penicillin concentration (P_0)	0 (g/L)
Substrate concentration (S_0)	55 (g/L)
Initial culture volume (V_0)	100 (L)
Feed flow rate (F)	0.1728 (L/h)

$$\frac{dX(t)}{dt} = \left(\frac{u_X S(t) X(t)}{K_X X(t) + S(t)} \right) - \frac{X(t)}{V(t)} \frac{dV(t)}{dt}, \tag{31}$$

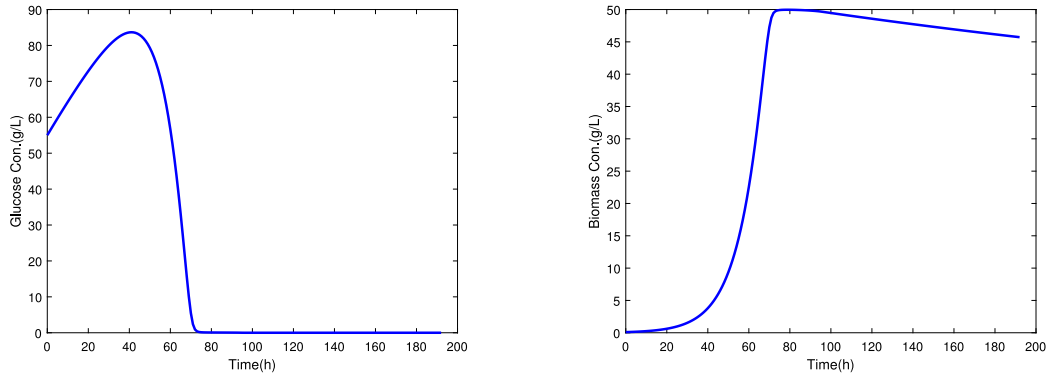
$$\frac{dP(t)}{dt} = \left(\frac{u_P S(t) X(t)}{K_P + S(t) + \frac{S^2(t)}{K_I}} \right) - K_H P(t) - \frac{P(t)}{V(t)} \frac{dV(t)}{dt}, \tag{32}$$

$$\frac{dS(t)}{dt} = -\left(\frac{1}{Y_{X/S}} \frac{u_X S(t) X(t)}{K_X X(t) + S(t)} \right) - \left(\frac{1}{Y_{P/S}} \frac{u_P S(t) X(t)}{K_P + S(t) + \frac{S^2(t)}{K_I}} \right) - m_X X(t) + \frac{F S_f}{V(t)} - \frac{S(t)}{V(t)} \frac{dV(t)}{dt}, \tag{33}$$

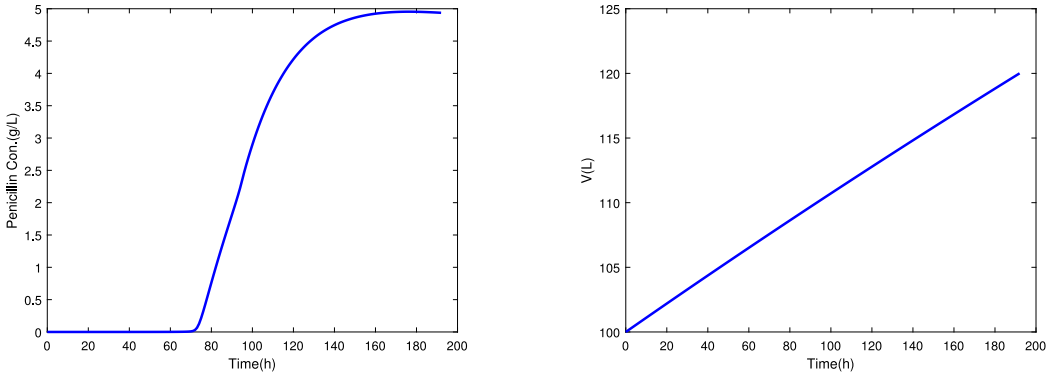
$$\frac{dV(t)}{dt} = F - 6.226 \times 10^{-4} V(t), \tag{34}$$

where X, P, S and V are biomass concentration, penicillin concentration, substrate concentration and culture volume, respectively. K_X and K_P are the biomass and penicillin saturation constants, respectively. K_I is the substrate inhibition constant of penicillin growth, K_H is the rate constant of penicillin hydrolysis. u_X and u_P are the biomass and penicillin specific growth rates, respectively. $Y_{X/S}$ and $Y_{P/S}$ are the biomass yield and penicillin yield per unit mass of substrate, respectively. m_X is the substrate maintenance coefficient, F is the substrate feeding rate, and S_f is the feed substrate concentration. The initial operating conditions and parameter values for the process simulator are shown in Table 1 and Table 2.

The sampling time for the penicillin fermentation process is chosen as 1 h. The batch time for production in the process simulator is selected as $t_f = 192h$ hours under the constraint condition $V \le 120L$ for the reactor volume. Fig. 3 shows the dynamic variations of glucose concentration, biomass concentration, penicillin concentration, and culture volume under the initial operating conditions given in Table 1. From the trends in concentration changes, it can be observed that the penicillin fermentation process undergoes the biomass growth phase and penicillin synthesis phase within the production batch. Therefore, to partition the growth stages using the algorithm described in Section 2, set $r = 2$, that is, clustering centers $c_j (j = 1, 2)$. The results of the growth



(a) Glucose (b) Biomass

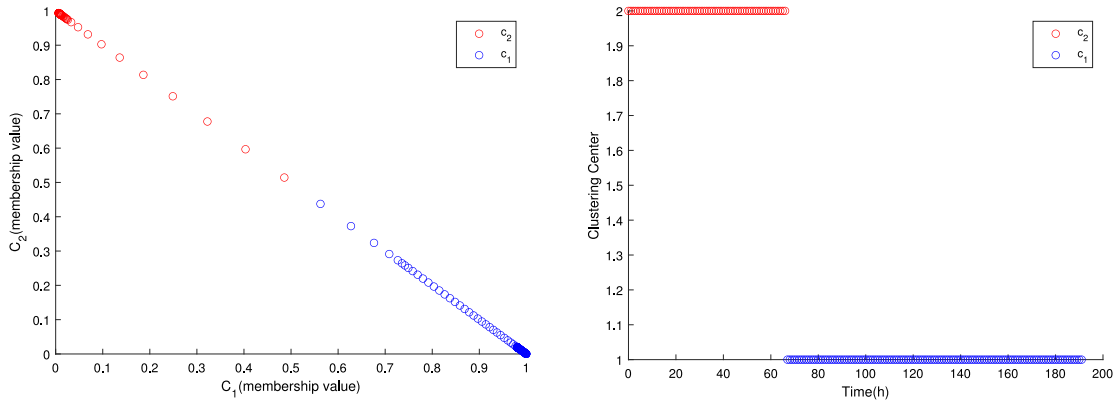


(c) Penicillin (d) Culture

Fig. 3. Dynamic change curves of the Glucose concentration, Biomass concentration, Penicillin concentration, and Culture medium volume.

Table 2
Initial model parameter values for the process simulator.

u_X	K_X	u_P	K_P	K_I	K_H	$Y_{X/S}$	$Y_{P/S}$	m_X	S_f
0.092	0.15	0.005	0.0002	0.1	0.04	0.45	0.9	0.014	600



(a) Membership value (b) Growth stage

Fig. 4. Division of each growth stage of the penicillin fermentation process.

stage partitioning are shown in Fig. 4. The biomass growth phase spans from 0–66 h, while the penicillin synthesis phase spans from 66–192 h.

To illustrate the impact of incorporating physical information on the ability of data-driven models to capture the dynamic characteristics of the fermentation process, this paper assumes that the dynamic characteristics of biomass concentration changes in the penicillin fermentation process are known. Eq. (29) can be expressed as:

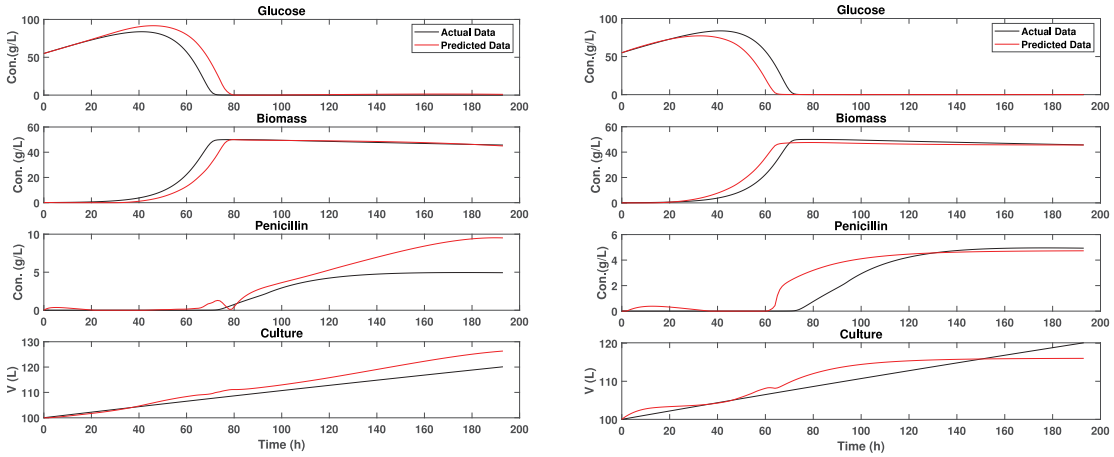
$$\frac{dx^*(t)}{dt} = \left(\frac{u_X S(t) X(t)}{K_X X(t) + S(t)}\right) - \frac{X(t)}{V(t)}\frac{dV(t)}{dt}. \tag{35}$$

Using the PINN-based multi-stage Koopman modeling method described in Section 3, the penicillin fermentation process is modeled

Table 3
The mean squared error.

Method	MSE				WCV
	Glucose	Penicillin	Biomass	Culture	Culture
Koopman modeling (DNN)	98.28	4.44	17.74	12.06	YES
Koopman modeling (PINN)	97.36	0.90	16.53	5.01	NO
Multi-stage Koopman modeling (DNN)	323.71	1.60	67.92	0.44	NO
Multi-stage Koopman modeling (PINN)	54.60	0.09	12.40	2.06	NO

"YES" indicates that the constraint is violated, while "NO" indicates that the constraint is not violated.



(a) Koopman Modeling based on General DNN (b) Koopman Modeling based on PINN

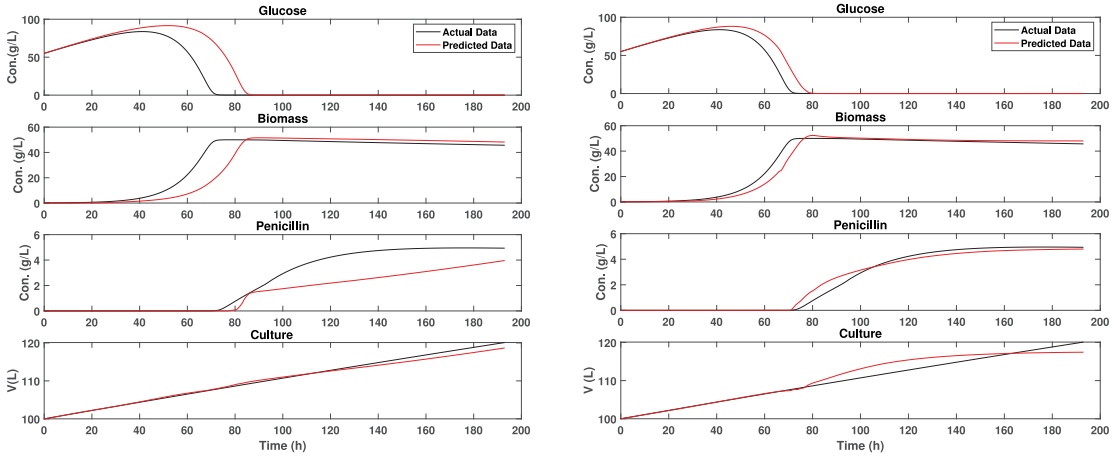
Fig. 5. Comparison between Koopman Modeling based on General DNN and Koopman Modeling based on PINN.

based on the partitioning results of the growth stages. We use the simulator to generate 100 batches of data within a 10% range of the initial conditions listed in Table 1 to train the model. To comprehensively evaluate the advantages of introducing PINN and multi-stage methods over existing Koopman modeling based on general DNN research, we designed the following comparative experiments: Koopman modeling based on PINN by introducing PINN into Koopman modeling, multi-stage Koopman modeling based on general DNN with the introduction of the multi-stage approach, and multi-stage Koopman modeling based on PINN by jointly introducing PINN and the multi-stage approach. The fitting performance of the models established by each modeling method is shown in Figs. 5 and 6. To more intuitively compare the performance of each model, we used two evaluation indicators: mean squared error (MSE) and whether the constraint is violated (WCV). The specific results are presented in Table 3.

From Table 3, it is evident that the Koopman modeling based on general DNN not only violates the volume constraint but also has a higher MSE. Incorporating process stage characteristics and physical information during the modeling process can effectively prevent constraint violations and reduce modeling errors. By combining these findings with the prediction results shown in Figs. 5 and 6, we can provide a detailed analysis of the effectiveness and advantages of the four modeling methods. The prediction errors for each model are most pronounced during the 20–70 h period when microbial growth is rapid, while the errors decrease after 70 h as microbial growth stabilizes. Specifically, due to the excessive sensitivity of the DNN model to input changes and the instability during the optimization process, the Koopman modeling method based on general DNN shown in Fig. 5(a) struggles to accurately describe the system's physical behavior due to the lack of physical constraints, leading to significant prediction errors and violations of volume constraints. In contrast, the Koopman modeling method based on PINN shown in Fig. 5(b) improves the model's physical consistency by incorporating knowledge of microbial fermentation dynamics. The physical consistency enhances the overall accuracy of the model and prevents violations of process constraints. However, the single-stage model may not fully capture the complex dynamic characteristics of the fermentation process, particularly during the penicillin synthesis phase. The multi-stage Koopman modeling method based on general DNN shown in Fig. 6(a) better captures the dynamic characteristics of different growth stages by modeling each stage of microbial fermentation separately, thereby avoiding violations of process constraints. However, the DNN model in each stage still lacks physical constraints, which can lead to overly sharp responses during stage transitions, resulting in significant prediction errors, especially during the penicillin synthesis phase. In contrast, the multi-stage Koopman modeling method based on PINN shown in Fig. 6(b) combines the advantages of physical constraints and multi-stage modeling. By incorporating physics-informed neural networks at each stage, the method effectively reduces prediction errors and more accurately simulates the dynamic changes in the fermentation process, providing smoother and more accurate predictions.

5. Conclusion

This paper proposes a multi-stage Koopman modeling method based on PINN. The method enables the dynamic mapping of microbial fermentation processes into high-dimensional linear space and the separate modeling of characteristics specific to different microbial growth stages. Specifically, we incorporate PINN into the approximation process of the Koopman operator for the first time to enhance the model's physical consistency and predictive performance. Results from the penicillin fermentation case demonstrate significant improvements achieved by our proposed method in modeling accuracy and practicality. Our method offers new theoretical and methodological support for the control and optimization of microbial fermentation processes. Additionally, the proposed algorithm framework integrates physical information with data-driven methods and demonstrates good scalability. It can be applied to other fields as long as the relevant field-specific physical information is incorporated.



(a) Multi-stage Koopman modeling based on general DNN (b) Multi-stage Koopman modeling based on PINN

Fig. 6. Comparison between multi-stage Koopman modeling based on general DNN and multi-stage Koopman modeling based on PINN.

CRediT authorship contribution statement

Quan Li: Writing – original draft, Methodology, Funding acquisition, Formal analysis, Conceptualization. **Jingran Zhang:** Methodology, Formal analysis. **Haiying Wan:** Writing – review & editing, Supervision, Funding acquisition. **Zhonggai Zhao:** Writing – review & editing, Supervision. **Fei Liu:** Writing – review & editing, Supervision, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgments

The authors acknowledge the support of the National Natural Science Foundation of China (NSFC 62403215, 61833007), the Natural Science Foundation of Jiangsu Province (BK20241607), the Postgraduate Research & Practice Innovation Program of Jiangsu Province (KYCX23_2441), and the Fundamental Research Funds for the Central Universities China (JUSRP123064).

References

[1] A.J. Straathof, S.A. Wahl, K.R. Benjamin, R. Takors, N. Wierckx, H.J. Noorman, Grand research challenges for sustainable industrial biotechnology, Trends Biotechnol. 37 (10) (2019) 1042–1050.
[2] A. Narasingam, J.S. Kwon, Koopman Lyapunov-based model predictive control of nonlinear chemical process systems, AIChE J. 65 (11) (2019) e16743.
[3] O. Levenspiel, The monod equation: a revisit and a generalization to product inhibition situations, Biotechnol. Bioeng. 22 (8) (1980) 1671–1687.
[4] S. Lopez, J. France, W. Gerrits, M. Dhanoa, D. Humphries, J. Dijkstra, A generalized michaelis-menten equation for the analysis of growth, J. Anim. Sci. 78 (7) (2000) 1816–1828.
[5] R.K. Bajpai, M. Reuss, A mechanistic model for penicillin production, J. Chem. Technol. Biotechnol. 30 (1) (1980) 332–344.
[6] S. Goldrick, C.A. Duran-Villalobos, K. Jankauskas, D. Lovett, S.S. Farid, B. Lennox, Modern day monitoring and control challenges outlined on an industrial-scale benchmark fermentation process, Comput. Chem. Eng. 130 (2019) 106471.
[7] B. de Andrés-Toro, J. Giron-Sierra, J. Lopez-Orozco, C. Fernandez-Conde, J.M. Peinado, F. Garcıéa-Ochoa, A kinetic model for beer production under industrial operational conditions, Math. Comput. Simulation 48 (1) (1998) 65–74.
[8] K. Sakimoto, M. Kanna, Y. Matsumura, Kinetic model of cellulose degradation using simultaneous saccharification and fermentation, Biomass Bioenergy 99 (2017) 116–121.
[9] O.J. Fisher, N.J. Watson, J.E. Escrig, R. Witt, L. Porcu, D. Bacon, M. Rigley, R.L. Gomes, Considerations, challenges and opportunities when developing data-driven models for process manufacturing systems, Comput. Chem. Eng. 140 (2020) 106881.
[10] M.S.F. Bangi, A. Narasingam, P. Siddhamshetty, J.S. Kwon, Enlarging the domain of attraction of the local dynamic mode decomposition with control technique: Application to hydraulic fracturing, Ind. Eng. Chem. Res. 58 (14) (2019) 5588–5601.
[11] M. Korda, I. Mezić, Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control, Automatica 93 (2018) 149–160.
[12] A. Narasingam, J.S. Kwon, Application of koopman operator for model-based control of fracture propagation and proppant transport in hydraulic fracturing operation, J. Process Control 91 (2020) 25–36.
[13] D. Bruder, X. Fu, R.B. Gillespie, C.D. Remy, R. Vasudevan, Data-driven control of soft robots using koopman operator theory, IEEE Trans. Robot. 37 (3) (2020) 948–961.
[14] B. Lusch, J.N. Kutz, S.L. Brunton, Deep learning for universal linear embeddings of nonlinear dynamics, Nat. Commun. 9 (1) (2018) 4950.
[15] Y.Q. Xiao, X.L. Zhang, X. Xu, X.Q. Liu, J.H. Liu, Deep neural networks with koopman operators for modeling and control of autonomous vehicles, IEEE Trans. Intell. Veh. 8 (1) (2022) 135–146.
[16] L.J. Feng, C.H. Zhao, Fault description based attribute transfer for zero-sample industrial fault diagnosis, IEEE Trans. Ind. Inform. 17 (3) (2020) 1852–1862.
[17] T.C. Zhang, J.L. Chen, S.L. He, Z.T. Zhou, Prior knowledge-augmented self-supervised feature learning for few-shot intelligent fault diagnosis of machines, IEEE Trans. Ind. Electron. 69 (10) (2022) 10573–10584.
[18] D.D. Zhang, E.A. Del Rio-Chanona, P. Petsagkourakis, J. Wagner, Hybrid physics-based and data-driven modeling for bioprocess online simulation and optimization, Biotechnol. Bioeng. 116 (11) (2019) 2919–2930.
[19] S. Espinel-Ríos, J. Avalos, Hybrid physics-informed metabolic cybergenetics: process rates augmented with machine-learning surrogates informed by flux balance analysis, Ind. Eng. Chem. Res. 63 (15) (2024) 6685–6700.
[20] G.E. Karniadakis, I.G. Kevrekidis, L. Lu, P. Perdikaris, S.F. Wang, L. Yang, Physics-informed machine learning, Nat. Rev. Phys. 3 (6) (2021) 422–440.
[21] A. Maksakov, S. Palis, Koopman-based data-driven control for continuous fluidized bed spray granulation with screen-mill-cycle, J. Process Control 103 (2021) 48–54.
[22] K. Azizzadenesheli, N. Kovachki, Z.Y. Li, M. Liu-Schiaffini, J. Kossaifi, A. Anandkumar, Neural operators for accelerating scientific simulations and design, Nat. Rev. Phys. (2024) 1–9.
[23] M.S.F. Bangi, K. Kao, J.S. Kwon, Physics-informed neural networks for hybrid modeling of lab-scale batch fermentation for β-carotene production using Saccharomyces cerevisiae, Chem. Eng. Res. Des. 179 (2022) 415–423.
[24] T.Q. Cui, T. Bertalan, N. Ndahiro, P. Khare, M. Betenbaugh, C. Maranas, I.G. Kevrekidis, Data-driven and physics informed modeling of Chinese Hamster Ovary cell bioreactors, Comput. Chem. Eng. 183 (2024) 108594.
[25] C.H. Zhao, F.L. Wang, N.Y. Lu, M.X. Jia, Stage-based soft-transition multiple PCA modeling and on-line monitoring strategy for batch processes.

[26] M. Asvad, A. Hajinezhad, A. Jafari, S.F. Moosavian, Multiscale kinetic modeling for biohydrogen production: a study on membrane bioreactors, Int. J. Hydrog. Energy 48 (76) (2023) 29641–29650.
[27] Y.Y. Hui, X.Q. Zhao, Multi-phase batch process monitoring based on multiway weighted global neighborhood preserving embedding method, J. Process Control 69 (2018) 44–57.
[28] Z.G. Su, P.H. Wang, J. Shen, Y.F. Zhang, L. Chen, Convenient T–S fuzzy model with enhanced performance using a novel swarm intelligent fuzzy clustering technique, J. Process Control 22 (2012) 108–124.
[29] S. Askari, Fuzzy C-means clustering algorithm for data with unequal cluster sizes and contaminated with noise and outliers: Review and development, Expert Syst. Appl. 165 (2021) 113856.
[30] D. Alfredo, S.J. Marco, Fuzzy modeling of a nonlinear continuous fermentation bioreactor for ethanol production, in: Asme International Mechanical Engineering Congress & Exposition, 2012, p. 941.
[31] B. Wang, M.Y. He, X.Y. Wang, H.Y. Tang, X.L. Zhu, A multi-model predictive control method for the pichia pastoris fermentation process based on relative error weighting algorithm, Alex. Eng. J. 61 (12) (2022) 9649–9660.
[32] B.O. Koopman, Hamiltonian systems and transformation in Hilbert space, Proc. Natl. Acad. Sci. USA 17 (5) (1931) 315–318.
[33] Q. Jiang, X. Yan, H. Yi, F. Gao, Data-driven batch-end quality modeling and monitoring based on optimized sparse partial least squares, IEEE Trans. Ind. Electron. 67 (5) (2019) 4098–4107.