Chemometrics and Intelligent Laboratory Systems 152 (2016) 88–96

Contents lists available at ScienceDirect

Chemometrics and Intelligent Laboratory Systems

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

A new data-driven modeling method for fermentation processes

Qiangda Yang [a], Hongbo Gao [b], Weijun Zhang [a], Zhongyuan Chi [a], Zhi Yi [a]

[a] School of Metallurgy, Northeastern University, Shenyang 110819, China
[b] School of Mechanical Engineering and Automation, Northeastern University, Shenyang 110819, China

ARTICLE INFO

Article history:
Received 12 March 2015
Received in revised form 8 July 2015
Accepted 19 January 2016
Available online 27 January 2016

Keywords:
Data-driven modeling
Artificial neural networks
Particle swarm optimization
Fermentation process
Nosiheptide

ABSTRACT

An accurate model is the premise for successfully implementing fermentation process optimization. Most data-driven models that are widely applied to fermentation processes are unfit for optimization or provide low precision. This paper presents a new data-driven modeling method for directly developing an ANN-based differential model that is fit for optimization. Moreover, this model can provide high precision because it can be discretized using the sampling period of the control variables as the step length. The lack of data pairs is addressed by transforming the model-training problem into a dynamic system parameter identification problem. Further, a particle swarm optimization algorithm with a time-varying escape mechanism (PSOE) is constructed to determine the model parameters. Finally, the uniform design method is used to select the model structure. The results of experiments conducted using practical data for a lab-scale nosiheptide batch fermentation process confirm the effectiveness of the proposed modeling method and PSOE algorithm.

1. Introduction

Recent decades have witnessed the widespread use of fermentation products in pharmaceuticals, chemicals, foods, energy resources, and environment protection agents [1–4]. However, in general, such products are produced in small amounts at high manufacturing costs. This problem can be mitigated by the optimization of fermentation processes (i.e., optimal control of fermentation processes) [5,6], which requires an accurate process model for successful implementation.

The three modeling paradigms currently available for fermentation processes are mechanistic modeling [7,8], data-driven modeling [9,10], and hybrid modeling [11,12]. Mechanistic modeling and hybrid modeling methods require certain a priori knowledge about the process being modeled; therefore, these modeling methods have some limitations with regard to fermentation processes for which sufficient a priori knowledge is not available. In contrast, data-driven modeling methods can develop the model of a fermentation process exclusively from its historical production data, with practically no a priori knowledge of the process; hence, such modeling methods find wider use than mechanistic and hybrid modeling methods. Existing data-driven modeling methods can be broadly classified into static modeling methods and dynamic modeling methods. Static modeling methods commonly develop the model of a fermentation process using current control variables (e.g., temperature and pH) and current non-control variables (e.g., carbon dioxide content of exhaust gases and oxygen uptake rate) as inputs, and current state variables (e.g., biomass concentration and substrate concentration) as outputs [13–16]. Because static models involve non-control variables, they are usually employed for soft sensing or process monitoring; however, they are unfit for process optimization. On the other hand, dynamic modeling methods commonly develop the model of a fermentation process using past control variables and past state variables as inputs, and current state variables as outputs [17–21]. Thus, such dynamic models, in a certain sense, can be regarded as data-driven mechanistic models expressed in discrete form. They can be used not only for soft sensing and process monitoring but also for process optimization. However, the long sampling period of state variables (a few hours or more) inevitably degrades the discretization precision, and thus, the model precision; consequently, optimization based on such models is unsatisfactory or may even fail.

This paper proposes a new data-driven modeling method for fermentation processes that directly develops a data-driven differential model for fermentation process optimization; hereafter, we refer to this method as the data-driven differential modeling method. The developed differential model can achieve high prediction precision via discretization using the sampling period of the control variables as the step length. Artificial neural networks have been widely employed for effective data-driven modeling of various processes [22–24]. To develop a fermentation process model, this study employs feed-forward neural networks (FNNs), which have been successfully applied to various fermentation processes [18,19,25,26], because they can approximate nonlinear relationships arbitrarily well [27]. Overcoming the lack of model target outputs is a major challenge for the data-driven differential modeling of fermentation processes. To this end, we transform the model-training problem into a dynamic system parameter identification problem in the present study.

The transformed parameter identification (TPI) problem is a complex high-dimensional optimization problem. In general, it is difficult to calculate the derivatives required by traditional optimization

E-mail address: yangqd@smm.neu.edu.cn (Q. Yang).

http://dx.doi.org/10.1016/j.chemolab.2016.01.013
0169-7439/

algorithms such as Newton's algorithm [28] and the least-squares algorithm [29]. Hence, artificial intelligence (AI)-based optimization algorithms such as the differential evolution (DE) algorithm [30], genetic algorithm (GA) [31], and particle swarm optimization (PSO) [32] are preferred for solving the TPI problem because they are applicable to high-dimensional problems, do not require derivative information for the objective function, and can perform global optimization [33]. In particular, PSO exhibits useful characteristics such as simple structure, ease of use, and fast convergence; therefore, it has been widely adopted for various purposes such as function optimization, neural network training, and parameter identification [34–36]. However, standard PSO (SPSO) can easily fall into local optima, especially in the case of complex high-dimensional problems [37] such as the TPI problem. In the present study, an improved PSO algorithm, namely PSOE, is developed by applying a time-varying escape mechanism to SPSO. Then, PSOE is employed to solve the TPI problem in order to obtain the model parameters, i.e., the weights and thresholds of the corresponding neural networks. In addition, the selection of the model structure, i.e., the topology of the corresponding neural networks, is discussed in detail.

The remainder of this paper is organized as follows: Section 2 describes the data-driven differential modeling method based on FNNs and PSOE, including the basic concept of data-driven differential modeling, the PSOE algorithm, the selection of the model structure, and the determination of model parameters. Section 3 discusses the application of the proposed modeling method to a lab-scale nosiheptide batch fermentation process, and presents the obtained results. Finally, Section 4 summarizes our findings and concludes the paper with a brief discussion on the scope for future studies.

2. Data-driven differential modeling of fermentation processes using FNNs and PSOE

2.1. Data-driven differential modeling method

Fermentation process optimization requires a model that describes the functional relationships between control variables and state variables. In general, according to fermentation kinetics and the mass balance principle, the functional relationships between control variables and state variables can be described by the following set of differential equations [38,39].

$$\begin{cases} \dot{x}_1(t) &= f_1\left(\boldsymbol{v}_1^x(t), \boldsymbol{v}_1^u(t)\right) \\ \dot{x}_2(t) &= f_2\left(\boldsymbol{v}_2^x(t), \boldsymbol{v}_2^u(t)\right) \\ \quad\vdots & \quad\vdots \\ \dot{x}_m(t) &= f_m\left(\boldsymbol{v}_m^x(t), \boldsymbol{v}_m^u(t)\right) \\ \quad\vdots & \quad\vdots \\ \dot{x}_M(t) &= f_M\left(\boldsymbol{v}_M^x(t), \boldsymbol{v}_M^u(t)\right) \end{cases} \tag{1}$$

where x_m ($m = 1, 2, \cdots, M$) are the elements of the state vector $\boldsymbol{x}$; f_m ($m = 1, 2, \cdots, M$) are the corresponding functions; $\boldsymbol{v}_m^x$ denotes a vector consisting of some state variables that are related to f_m; and $\boldsymbol{v}_m^u$ denotes a vector consisting of some control variables that are related to f_m.

Owing to the inherent complexity of fermentation processes, it is usually difficult to derive concrete expressions for the functions f_m ($m = 1, 2, \cdots, M$) from first-principles calculations. Therefore, the proposed data-driven differential modeling method approximates these functions using FNNs as follows.

$$\begin{cases} \dot{x}_1(t) &= \hat{f}_1\left(\boldsymbol{v}_1^x(t), \boldsymbol{v}_1^u(t), \boldsymbol{\theta}_1\right) \\ \dot{x}_2(t) &= \hat{f}_2\left(\boldsymbol{v}_2^x(t), \boldsymbol{v}_2^u(t), \boldsymbol{\theta}_2\right) \\ \quad\vdots & \quad\vdots \\ \dot{x}_m(t) &= \hat{f}_m\left(\boldsymbol{v}_m^x(t), \boldsymbol{v}_m^u(t), \boldsymbol{\theta}_m\right) \\ \quad\vdots & \quad\vdots \\ \dot{x}_M(t) &= \hat{f}_M\left(\boldsymbol{v}_M^x(t), \boldsymbol{v}_M^u(t), \boldsymbol{\theta}_M\right) \end{cases} \tag{2}$$

where $\hat{f}_m(m = 1, 2, \cdots, M)$ denote the FNN-based sub-models used to approximate the unknown functional relationships f_m ($m = 1, 2, \cdots, M$), and $\boldsymbol{\theta}_m$ ($m = 1, 2, \cdots, M$) denote the parameter vectors corresponding to the m sub-models, i.e., the weights and thresholds of the corresponding FNNs.

However, for practical fermentation processes, the available modeling data include only the measured data for the state and control variables; hence, only inputs $[\boldsymbol{v}_m^x(t), \boldsymbol{v}_m^u(t)]$ are available, whereas no target outputs $\dot{x}_m(t)$ are available to train the sub-models $\hat{f}_m(m = 1, 2, \cdots, M)$. Thus, these sub-models cannot be trained using conventional methods. Therefore, the proposed data-driven differential modeling method trains each sub-model in the following two steps. First, $\boldsymbol{\theta}_m$ ($m = 1, 2, \cdots, M$) are regarded as parameter vectors to be identified in the dynamic system described by (2); thus, the training procedure of the sub-models is transformed into a dynamic system parameter identification problem. Second, $\boldsymbol{\theta}_m$ ($m = 1, 2, \cdots, M$) are determined by minimizing the objective function defined in (3) using only the measured data for the state and control variables; consequently, all sub-models are trained without the need for target outputs. The objective function is defined as

$$J(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \cdots, \boldsymbol{\theta}_m, \cdots, \boldsymbol{\theta}_M) = \frac{\sum_{b=1}^{B}\sum_{h=1}^{H_b}\sum_{m=1}^{M}\left|\frac{x_{bhm} - \hat{x}_{bhm}}{x_{bhm}}\right|}{M \cdot \sum_{b=1}^{B} H_b} \tag{3}$$

where b denotes the bth batch of training data; B is the number of batches; h denotes the hth set of offline measurements of state variables; H_b is the number of state variable samples in batch b; m denotes the mth state variable; M is the number of state variables; and x and $\hat{x}$ denote the measured and predicted values, respectively.

There are two main concerns with regard to the development of an accurate model for a fermentation process using the data-driven differential modeling method. The first one is how to solve the parameter identification problem of the dynamic system described by (2) and then obtain the parameters of each sub-model (i.e., the weights and thresholds of the corresponding FNN). The second one is how to select a suitable structure for each sub-model (i.e., the topology of the corresponding FNN).

In the following sub-sections, first, a PSOE algorithm with a time-varying escape mechanism is proposed. Then, the proposed algorithm is employed to determine the parameters of each sub-model. Finally, the structure of each sub-model is selected on the basis of the uniform design method and the leave-one-out cross-validation technique.

2.2. PSOE

2.2.1. SPSO algorithm

PSO is a population-based heuristic optimization algorithm originally proposed by Kennedy and Eberhart [40]; it was inspired by the flocking behavior of birds and schooling behavior of fish [41]. PSO involves a swarm of particles that collectively move in search of the global optimum. The particles search for the optimal solution by moving in the D-dimensional search space, and the position of each particle s, which can be represented by a D-dimensional vector $\boldsymbol{p}_s = (p_{s1}, p_{s2}, \cdots, p_{sD})$, corresponds to a candidate solution of the optimization problem at hand, where $s = 1, 2, \cdots, S$, and S denotes the swarm size. Initially, the particles are randomly placed at different positions in the search space. Each particle is assigned a fitness value that represents its performance on the objective function of the problem. Next, an iterative process begins, whereby the particles collectively move in the search space. In particular, based on some random permutations, each particle follows both its own best position that it has discovered, denoted by $\boldsymbol{pbest}_s = (pbest_{s1}, pbest_{s2}, \cdots, pbest_{sD})$, as well as the best position discovered by the entire swarm thus far, denoted by $\boldsymbol{gbest} = (gbest_1, gbest_2, \cdots,$