Neurocomputing 148 (2015) 216–221

Contents lists available at ScienceDirect

Neurocomputing

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

The probability density function based neuro-fuzzy model and its application in batch processes

Li Jia*, Kai Yuan

Shanghai Key Laboratory of Power Station Automation Technology, Department of Automation, College of Mechatronics Engineering and Automation, Shanghai University, Shanghai 200072, China

ARTICLE INFO

Article history:
Received 14 March 2013
Received in revised form
9 September 2013
Accepted 14 September 2013
Available online 28 June 2014

Keywords:
Batch process
Probability density function
Neuro-fuzzy model
Modeling error

ABSTRACT

Motivated by the concept of probability density function (PDF) control, a new probability density function (PDF) based neuro-fuzzy model for batch processes is proposed in this paper. The probability density function (PDF) of modeling error is introduced as a criterion to measure the performance of the neuro-fuzzy model of batch processes. More specifically, the neuro-fuzzy model parameter updating approach is transformed into the shape control of the probability density function (PDF) of the modeling error. That is to say, the PDF shape control idea is used to tune neuro-fuzzy model parameters so that the modeling error PDF is controlled to follow a targeted PDF, which is Gaussian or uniform distribution. As a result, the mean square error and the distribution of modeling error are both considered. Moreover, it alternatively uses the method of minimum-entropy to acquire the parameters of the neuro-fuzzy model if the targeted probability density function (PDF) is unknown. An example is applied to illustrate the applicability of the proposed method and the simulation results show that the proposed approach is more effective.

1. Introduction

Batch processes have been widely used in the production of low volume and high value added products due to their special characteristics [1–3]. It is important to optimize the control operation trajectory to derive the maximum benefit from batch processes, which depends on obtaining an accurate model. However, building the first principle model for batch processes usually costs a lot of time and effort.

With the development of information science, large amount of data can be collected and stored in the computer during the production process, which imply that the internal information of mechanism, equipment operation and changes in the production process. However, the input–output data points collected from batch processes are normally insufficient due to the limited batch runs and production time when manufacturing a particular product. Recently, neural networks have been suggested to identify batch processes because of its powerful ability to approximate a nonlinear function to any arbitrary accuracy [4–10]. And the focus of attention is to design the identification algorithm to enforce that the predicted output of neural network converges to the actual one. That is to say, the identification algorithm method is to identify the parameters of neural network to ensure the output of the model to track the output of the practical batch process. In general, minimum square error (MSE) is employed as the criterion when building a neural network based model. That is to say, the identification algorithm is designed to minimize the mean of squared errors of neural network based model. If the probability density function (PDF) of the modeling error is Gaussian, it is equivalent to build a neural network to assure the distribution of the modeling error as narrower as possible. In essence, this kind of algorithm is just from the viewpoint of a single sample point but not the distribution of data points of different operation domain. However, the input–output data points collected from batch process are normally corrupted by noise and thus the PDF of the modeling error is not Gaussian or non-uniform distribution [11–14]. As a result, above mentioned approach cannot guarantee the generalization and robustness of the model. Theory analysis and simulation result showed that the probability density function (PDF) of the modeling error should be considered when building a neural network based model for batch processes. In addition, the traditional neural network is essentially a black-box with unexplainable internal mapping so that it is difficult to use the repetitive mechanism knowledge of batch processes.

Considering above-mentioned problems, a new probability density function (PDF) based neuro-fuzzy model for batch processes is

* Corresponding author.
E-mail address: jiali@staff.shu.edu.cn (L. Jia).

http://dx.doi.org/10.1016/j.neucom.2013.09.068
0925-2312/

Nomenclature

k	batch index
t	time index
t_f	batch run length
y_d	specified reference trajectory
$u_k(t)$	the input of the tth time of the kth batch
$y_k(t)$	the output of the tth time of the kth batch
U_k	input sequence of the $k$thbatch
Y_k	output sequence of the kth batch
$\hat{Y}_k$	predicted output sequence at the kth batch
$\hat{y}_k(t)$	predicted output of the tth time of the kth batch
$\mu_j(x_k(t))$	Gaussian radial basis function
N	the number of fuzzy rules
c_{ji} and θ_j	centre and width of neuro-fuzzy model
w	consequent parameter of neuro-fuzzy model
$e_k(t)$	modeling error of the tth time of the kth batch
Γ_{error}	the PDF of the modeling error
Γ_{target}	the targetted PDF
$p(v)$	the probability density function
$\varphi(\bullet)$	window function
h_P	window width
σ_g	the factor of targeted PDF
J_1	criterion function of probability density function control system
r_1	step length of iteration
H	entropy
$l(\cdot)$	the probability mass function
J_2	criterion function of entropy control system
r_2	step length of iteration

proposed in this paper. It is motivated by the concept of the probability density function (PDF) control proposed by Wang [15]. The probability density function (PDF) of modeling error is introduced as a criterion to measure the performance of the neuro-fuzzy model of batch processes. More specifically, the neuro-fuzzy model parameter updating approach is transformed into the shape control of the probability density function (PDF) of the modeling error. That is to say, the PDF shape control idea is used to tune neuro-fuzzy model parameters so that the modeling error PDF is controlled to follow a targeted PDF, which is Gaussian or uniform distribution. As a result, the mean square error and the distribution of modeling error are both considered. In practice, the data used for neuro-fuzzy modeling always contains stationary noise, the targeted PDF should be selected as with zero mean but have a variance that is equivalent to the stationary variance of the data [14]. Moreover, it alternatively uses the method of minimum-entropy to acquire the parameters of the neuro-fuzzy model if the targeted probability density function (PDF) is unknown.

The rest of this paper is organized as follows. In Section 2, neuro-fuzzy model for batch process is presented. Section 3 introduces the identification of probability density function based neuro-fuzzy model. Simulation example is given in Section 4, followed by the conclusion given in Section 5.

2. Neuro-fuzzy model for batch processes

Set that $k(k=1,2,\ldots,K)$ and t_f as the kth batch and the end time of one batch, respectively. In the kth batch, the input and output of the tth time are represented by $u_k(t)$ and $y_k(t)$. In order to describe the problem clearly, suppose that one batch is divided into T equal intervals, namely $t=1,2,\ldots,T$. Thus, the input sequence of the kth batch is $U_k=[u_k(1),\ldots,u_k(T)]^T$ and the corresponding output sequence is $Y_k=[y_k(1),\ldots,y_k(T)]^T$. In this paper, a PDF-based neuro-fuzzy model (NFM) is proposed to identify the batch process as follows.

$$\hat{y}_k(t+1)=f_{\text{NFM}}(x_k(t)) \tag{1}$$

where

$$x_k(t)=\left[y_k(t),y_k(t-1),\ldots,y_k(t-n_y+1),u_k(t),u_k(t-1)\ldots,u_k(t-n_u+1)\right]^T$$

where $\hat{y}_k(t+1)$ is the predicted output of the $t+1$th time of the kth batch, and n_y and n_u are integers related to the model order. Then the predicted output sequence of the kth batch is

$$\hat{Y}_k=[\hat{y}_k(1),\ldots,\hat{y}_k(T)]^T \tag{2}$$

The proposed NFM model consists of five layers. The first layer is the input layer, which contains n_y+n_uneurons. The second layer is fuzzy layer and has $(n_y+n_u)\times N$ neurons, in which N denotes the number of fuzzy IF-THEN rules and each neuron in this layer represents a membership function. The third layer is the rule layer and consists of N neurons. The fourth layer is the fuzzy decision layer with two neurons. Finally, the fifth layer is output layer. The output of the NFM model is given by:

$$\text{Output}=\frac{\sum_{j=1}^{N} w_j\mu_j(x_k(t))}{\sum_{j=1}^{N}\mu_j(x_k(t))}=\frac{\sum_{j=1}^{N} w_j\times\exp\left(-\sum_{i=1}^{n_y+n_u}((x_k(t,i)-c_{ji})^2/\theta_j^2)\right)}{\sum_{j=1}^{N}\exp\left(-\sum_{i=1}^{n_y+n_u}((x_k(t,i)-c_{ji})^2/\theta_j^2)\right)} \tag{3}$$

where $\mu_j(x_k(t))$ denotes Gaussian radial basis function, w_j is the jth consequence of fuzzy rule, and c_{ji} and θ_j are centre and width, respectively.

A clustering algorithm in our previous work [16] is employed to estimate the antecedent parameters of N, c_{ji} and θ_j. Then the task of this work is to identify the parameters of $w=\{w_1,w_2,\ldots w_N\}$. The procedure is discussed in Section 3.

3. Identification of probability density function based neuro-fuzzy model

Motivated by the concept of PDF control, the parameters of the neuro-fuzzy model are identified by designing a virtual probability density function control system as shown in Fig. 1. The neuro-fuzzy model parameter updating approach is transformed into the shape control of the probability density function (PDF) of the modeling error. That is to say, the PDF shape control idea is used to tune neuro-fuzzy model parameters so that the modeling error PDF is controlled to follow a target PDF. The proposed procedure consists of two steps: (1) estimate PDF of the modeling error Γ_{error}; (2) design the PDF controller to minimize the difference between Γ_{error} and Γ_{target}.

3.1. Estimation of PDF of the modeling error

In this work, non-parametric Parzen windows method [17] is employed to estimate the PDF of the modeling error. It summarized as follows.

(1) Calculate the modeling error samples $\{e_k(t)\}$ $(k=1,2,\ldots,K\quad t=1,2,\ldots,T)$

$$e_k(t)=y_k(t)-\hat{y}_k(t) \tag{4}$$