Soft-sensor Modeling of SMB Chromatographic Separation Process Based on Incremental Extreme Learning Machine

Qing-Da Yang, Cheng Xing*, Jie-Sheng Wang, Yong-Cheng Sun, Yi-Peng ShuangGuan

Abstract—**Simulated Moving Bed (SMB) chromatography separation is an innovative technology that combines conventional fixed bed adsorption and true moving bed (TMB) chromatography separation techniques. Through analysis of the SMB chromatography isolation method, we have identified auxiliary variables for the soft-sensing model and key economic and technical indicators for the forecasting model. Our objective is to predict the component purity and yield of the elicit and residual solution in the SMB chromatographic separation process. To achieve this, we have utilized three different soft-sensing modeling methods: incremental extreme learning machine (I-ELM), inverse-free extreme learning machine (IF-ELM), and incremental regularized extreme learning machine (IR-ELM). Our simulation results demonstrate the effectiveness and accuracy of these ELM methods in predicting essential economic and technical gauges in the SMB chromatographic separation process. These methods enable instantaneous, optimized, and resilient operation of SMB chromatography separation. Overall, SMB chromatography separation represents an advanced technology that enhances traditional techniques, and our study highlights the efficacy of various ELM methods in predicting essential process indicators, thereby ensuring optimal operation.**

Index Terms—**SMB chromatography separation, Soft-sensor modeling, Incremental extreme learning machine**

I. Introduction

SMB chromatography chromatography separation is an innovative technology that combines conventional fixed bed adsorption and true moving bed (TMB) chromatography separation techniques [1]. Simulated Moving Bed (SMB) chromatographic separation technology is widely recognized as a highly efficient method in adsorption separation. It involves the use of multiple columns operated in a periodic manner to replicate counter-current flow between two phases, maintaining a continuous feed stream and product discharge. This technology has gained significant recognition across various industries, including chemical, biological, and food sectors, due to its exceptional continuity, low energy consumption, and high separation efficiency [2]. During continuous production, the SMB system operates in a periodic manner, which presents challenges in achieving optimal periodic stability while meeting performance objectives. For instance, product cleanliness, output, and usage of mobile phase. The complexity involved in the SMB chromatographic separation process, combined with the influence of various parameters and disturbances, further complicates the task of maintaining the optimal operating point over an extended period [3]. Accurately measuring component purity and yield poses a significant challenge in practical production settings, as limited detection devices and field constraints hinder the acquisition of real-time essential economic and technical metrics of chromatographic separation. As a result, achieving direct quality closed-loop control becomes a formidable task [4-5]. However, soft sensing technology emerges as a viable approach for effectively forecasting key indicators in complex industrial processes [6].

A comprehensive review conducted by Ref. [7] examined the identification of auxiliary variables for soft measurement models and significant economic and technical predictors in SMB chromatographic separation techniques. Ref. [8] proposed an adaptive soft-sensor modeling approach to predict the composition degree of purity in the extract and extraction solution trough-out the SMB chromatographic separation process. This method utilized a dynamic fuzzy neural network (D-FNN) and a moving window strategy, comparing algorithms such as Kalman filter (KF), linear least squares (LLS), and extended Kalman filter (EKF) with the generalized dynamic fuzzy neural network (GD-FNN) relying on soft-sensor. Ref. [9] introduced a soft-sensor modeling method employing improved particle population optimization (PSO) and minimum mean square (LMS) techniques. Building upon the foundations of the adaptive neural fuzzy inference system (ANFIS), a novel soft-sensor modeling approach was introduced by the authors in Ref. [10]. This approach was specifically designed to accurately forecast extract purity level and extract components in the context of SMB chromatographic separation techniques.

Manuscript received June 19, 2023; revised August 29, 2023. This work was supported by the Basic Scientific Research Project of Institution of Higher Learning of Liaoning Province (Grant No. LJKZ0307), and Postgraduate Education Reform Project of Liaoning Province (Grant No. LNYJG2022137).

Qing-Da Yang is an undergraduate student of School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, 114051, P. R. China (e-mail: 1486331702@qq.com).

Cheng Xing is a Ph.D candidate in School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, 114044, P. R. China (Corresponding author, phone: 86-0412-2538246; fax: 86-0412-2538244; e-mail: xingcheng0811@163.com).

Jie-Sheng Wang is a professor of School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, 114051, P. R. China (e-mail: wang_jiesheng@126.com).

Yong-Cheng Sun is an undergraduate student of School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, 114051, P. R. China (e-mail: 3359916640@qq.com).

Yi-Peng ShuangGuan is an undergraduate student of School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, 114051, P. R. China (e-mail: shangguan011130@126.com).

In 2006, Professor Huang from Nanyang Technological University in Singapore introduced Extreme Learning Machine (ELM) is an example of feed-forward neural network architecture [11-12]. The architecture of the ELM model resembles that of a BP neural network, but with a fixed number of layers, including the input stage, intermediate stage, and output stage. The input layer to hidden layer connections are initialized with random weights and offsets, whereas the weights associated with the connections from the hidden layer to the output layer are derived using the least squares method. ELM has gained popularity in various fields, such as face recognition [13], fault diagnosis [14], soft-sensor modeling [15], resilient setup and smart management of photovoltaic systems [16], on account of its swift learning time, efficient execution, and effective generalization power.

This article proposes a soft-sensing modeling technique for SMB chromatographic separation processes employing three incremental extreme learning machines. The article is organized as follows. Section 2 presents a comprehensive overview of SMB chromatographic separation technology and the structure of the soft-sensing model. Section 3 introduces the concept of extreme learning machines. Section 4 elaborates on the methodology of incremental extreme learning machines. Section 5 demonstrates the experimental simulation and result analysis. In conclusion, the paper concludes by summarizing the findings.

II. SMB Chromatographic Separation Technology and Soft-sensor Modeling

A. SMB Chromatographic Separation Technology

The Simulated Moving Bed (SMB) chromatographic separation technology aims to emulate the motion of the stationary phase adsorbent by constantly interchanging the positions of the feed and discharge ports. This technique utilizes a loop configuration consisting of a series of interconnected chromatographic columns.By sequentially shifting raw material inlet: the inlet for the raw material, and raffinate outlet in the direction of the mobile phase, SMB successfully emulates the counter-current flow between the mobile phase: the moving phase or the mobile component, resulting in the segregation of two components. Fig. 1 provides a visual representation of the fundamental principle behind SMB chromatographic separation [8].

In the given illustration, it assumes that constituent elements A and B are the two substances to be separated, with component A exhibiting a higher adsorption power than component B. The eluent, referred to as D, acts as the desorbent, while the extract, denoted as E, represents the desired product. The feed is labeled as F, and the raffinate as R. The entire bed can be divided into four zones (I, II, III, IV, or 1, 2, 3, 4) based on the sites of liquid entry and exit and their respective functionalities. Each zone serves a distinct purpose in the overall separation process.

B. Soft-sensing Model of SMB Chromatographic Separation Process

In the SMB chromatographic adsorption separation process, real-time measurement of the purity and component yield in the extraction product and residual solution is challenging. Moreover, various factors influence the variability of component purity and yield during the SMB separation process. Hence, the inauguration of a soft-sensor model for component purity and yield is of great theoretical significance and engineering applied significance. The soft-sensing model can be represented as follows:

$$\hat{X} = f(d, u, y, X^*, t) \tag{1}$$

where, $\hat{X}$ is the estimated worth of the model, d is the turbulence factor, u is the input control parameter, y is the response variable, X^* is the off-line sampling value of the estimated variable or the value calculated by analysis.

Fig. 2 presents a visual depiction of the layout of the soft-sensor model utilized in the SMB chromatographic separation process.

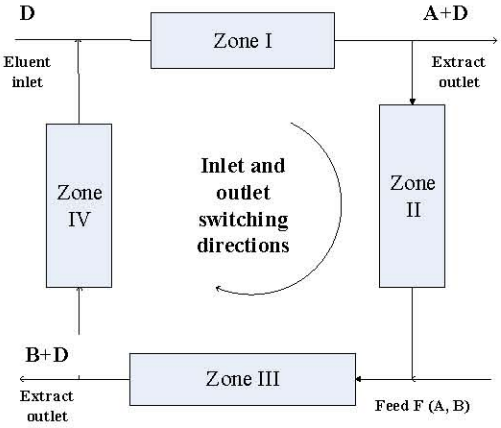

Fig. 1 Operating principle of SMB chromatographic separation process.

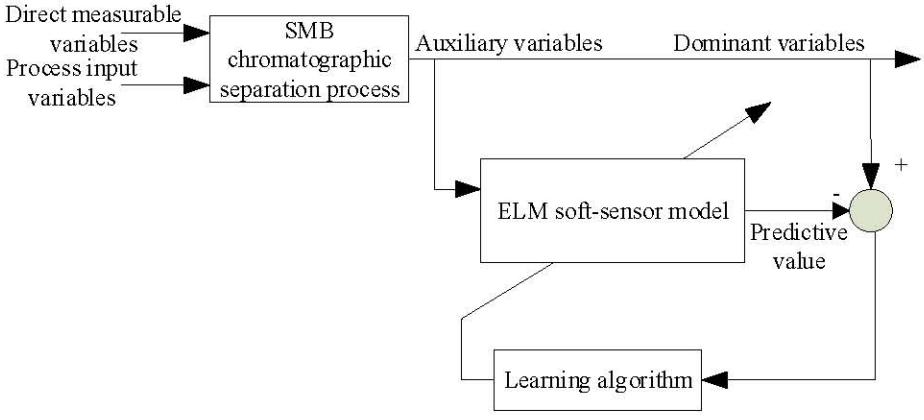

Fig. 2 Structure of soft-sensor model.

To develop the model, several auxiliary variables were selected guided by the process flow and preexisting knowledge. These variables are used to enhance the accuracy and reliability of the soft-sensor model.

(1) Fluid throughput of feed liquid injection inlet and injection pump (F pump), and unit is ml/min;

(2) Fluid throughput of the rinsing pump at the rinsing liquid inlet (D pump), and unit is ml/min;

(3) Time to switch the valve, and unit is min.

Determine the following variables as the output variables of the soft-sensor model.

(1) Integrity of the target substance in the effluent from port N (If there is an impurity effluent from port N, purity < 1);

(2) Degree of impurity in the outlet M (if there is a target outlet M, purity < 1);

(3) Mass recovery of the target object at port N/the injection mass of the target object can be obtained by the yield of the target object at port N;

(4) The quality of the impurities flowing out of port M/the quality of the impurities injected into the sample, to assess the productivity of the impurities at port M.

Table I shows the primary and the secondary variables incorporated within the soft-sensor model for the SMB chromatographic separation process. The auxiliary variables serve as inputs, while the outputs include the purity of the target component in port N, the purity of the impurity component in port M, the yield of the target component in port M, and the yield of the impurities in port M. These variables are employed to develop a forecasting model for the relevant economic and technical indicators utilizing the ELM neural network. To build the soft-sensor model, a dataset consisting of 1000 data points is utilized, as shown in Table II.

III. SMB Chromatographic Separation Technology and Soft-sensor Modeling

A. *Incremental Extreme Learning Machine(I-ELM)*

The ELM is an innovative Single-hidden-layer multi-layer perceptron that demonstrates similarities to the conventional BP neural network. The ELM neural network adopts a fully connected structure as its foundation, but three-layered architecture with input, hidden, and output layers. The Incremental Extreme Learning Machine (I-ELM) is a single hidden layer feed-forward neural network that provides advantages such as reduced training parameters and accelerated convergence rate.The structure of the I-ELM network is depicted in Fig. 3. has an m inputs and m outputs. $\{a_1, a_2, \cdots, a_m\}$ represents input weights of neurons in the hidden layer; b_L represents the threshold of the L neurons in the hidden layer; $\{\beta_1, \beta_2, \cdots, \beta_n\}$ is weight parameters from hidden layer to output layer; $X = [x_1, x_2, \cdots, x_m]$ is input representation; The dimension is $m \times N$; $Y = [y_1, y_2, \cdots, y_m]$ is output representation with dimension $n \times N$.

Install the maximum hidden unit quantity as M , the expected value error as ε , and the residual difference as E, that is, the variation matrix between the actual outcome generated by the network and the target. The number of hidden layer neurons L increases from 1. The steps to train I-ELM are described as follows.

Table I. Units and ranges of selected variables

Name	Injection pump flow capacity (F pump)	Injection pump flow capacity (D /pump)	Switch time	Purity of target substance in N port	Impurity purity in M port	Yield of target at port N	Yield of impurity at port M
Unit	ml/min	ml/min	min	mg/ml	mg/ml	%	%
Range	0-1	0-1	0-1	11-20	0-1	0-100	0-100

Table II. Data of SMB chromatographic separation process

Serial number	F pump flow rate (ml/min)	D pump flow rate (ml/min)	Switch time (min)	Purity of target substance in N port (%)	Impurity purity in M port (%)	Yield of target at port N (%)	Yield of impurity at port M (%)
1	0.15	0.50	11.00	41.25	24.96	12.53	64.20
2	0.15	0.50	12.00	85.11	34.21	32.16	88.75
3	0.15	0.50	13.00	97.38	42.46	92.03	97.35
4	0.15	0.50	14.00	99.55	51.24	63.61	99.42
5	0.15	0.50	15.00	99.92	59.75	74.12	99.89
6	0.15	0.50	16.00	99.99	69.69	83.27	99.98
7	0.15	0.50	17.00	100.00	84.55	92.97	100.00
8	0.15	0.50	18.00	100.00	95.32	98.11	100.00
9	0.15	0.50	19.00	100.00	99.37	99.75	100.00
…	…	…	…	…	…	…	…
1000	0.15	0.50	20.00	100.00	99.98	99.98	100.00

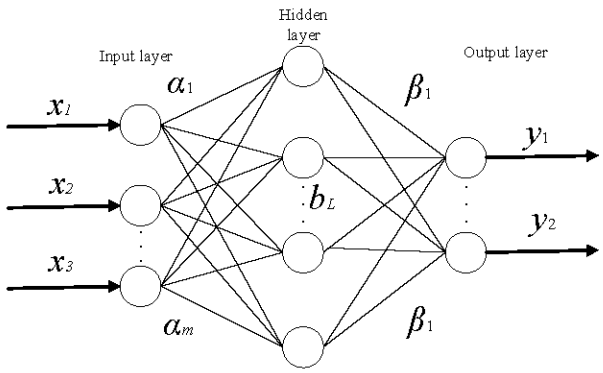

Fig. 3 I-ELM network structure.

When $L<\mathrm{M}$ and the error is greater than the expected error ε, there are the following steps.

(1) $L=L+1$.

(2) Input weight and threshold of intermediate layer units are randomly obtained.

(3) Calculate the input of intermediate layer units.

1) Addition of intermediate layer units. Extend to a matrix of (the same for each column), and then calculate:

$$x=aX+b \tag{2}$$

2) Radial hidden layer neurons. Extend to a matrix of (each row is the same) and then calculate:

$$x=b\left\|X^T-a\right\| \tag{3}$$

(4) Calculate the intermediate layer units.

Additive neuron output is derived by:

$$H=g(x)=\frac{1}{1+\exp(-x)} \tag{4}$$

Radial basal hidden layer neuron output is calculated by:

$$H=g^T(x) \tag{5}$$

(5) Compute the output weight of hidden layer neurons.

$$E'=E-\beta H \tag{6}$$

Compute the residual difference after adding E' hidden layer neuron, and repeat the above operations until the residual difference E' is less than ε, stop learning. If the error is consistently greater than the expected value, learning stops at $L>\mathrm{M}$.

$$\mathrm{H}=\frac{\mathrm{EH}^T}{\mathrm{HH}^T} \tag{7}$$

B. Inverse-free Extreme Learning Machine (IF-ELM)

In the model of extreme learning machine, the training error decreases with the increase of quantity of nodes in the hidden layer. But in the experiment, considering the computational complexity, quantity of hidden layer nodes should be reduced as far as possible. In order to balance the two factors of training error and computational complexity, it is urgent to determine the optimal value of the node number of hidden layer. The inverse-free extreme learning machine (IF-ELM) comes into being. It uses hidden layer nodes and increases the strategy. Node output weight coefficients with $L+1$ hidden layers can be calculated from the output weights of the nodes with L hidden layer nodes, without re-calculating the output weights of all hidden layer nodes.

When there are L hidden layer nodes, the hidden layer output matrix is H, and the output weight β^L can be expressed as:

$$\beta^L=H^{\varsigma}Y=\left(H^TY\right)^{-1}H^TY \tag{8}$$

When a hidden layer node is added, the output weight W^{L+1} and offset value E^{L+1} are updated to the following form.

$$\begin{aligned}W^{L+1}&=\left[W^L \quad w^{L+1}\right]\\ E^{L+1}&=\left[E^L \quad e^{L+1}\right]\end{aligned} \tag{9}$$

where, W^L and E^L are the output weight and deviation with L hidden layer nodes number; w^{L+1} is the output weight of the newly added hidden layer node, and e^{L+1} is the newly added hidden layer node and deviation, both of which are randomly selected parameters. Then the hidden layer output matrix of ELM with $L+1$ hidden layer nodes is H^{L+1} in Eq. (10).

$$\begin{aligned}H^{L+1}&=g\left(XW^{L+1}+E^{L+1}\right)\\&=g\left(X\left[W^L \quad w^{L+1}\right]+\left[E^L \quad e^{L+1}\right]\right)\\&=\left[H \quad h\right]\end{aligned} \tag{10}$$

where, $h=g\left(Xw^{L+1}+e^{L+1}\right)$. The calculation formula of output weight β^{L+1} of nodes with $L+1$ hidden layers is:

$$\begin{aligned}\beta^{L+1}&=\left(H^{L+1}\right)^{\varsigma}Y\\&=\left(\left[H \quad h\right]^{\mathrm{T}}\left[H \quad h\right]^{-1}\right)\left[H \quad h\right]^{\mathrm{T}}Y\\&=UY\end{aligned} \tag{11}$$

where, $\left(H^{L+1}\right)^{\varsigma}$ is the Moore-Penrose generalized matrix of H^{L+1}; $U=\left(\left[H \quad h\right]^{\mathrm{T}}\left[H \quad h\right]\right)^{-1}\left[H \quad h\right]^{\mathrm{T}}$.

In order to avoid over-fitting in learning, the regularization term a is added. Then:

$$\begin{aligned}U&=\left(a^2I_{L+1}+\left[H \quad h\right]^{\mathrm{T}}\left[H \quad h\right]\right)^{-1}\left[H \quad h\right]^{\mathrm{T}}\\&=\begin{bmatrix}a^2I_L+H^{\mathrm{T}}H & H^{\mathrm{T}}h\\ h^{\mathrm{T}}H & a^2+h^{\mathrm{T}}h\end{bmatrix}\\&=\begin{bmatrix}A & B\\ C & D\end{bmatrix}\begin{bmatrix}H^{\mathrm{T}}\\ h^{\mathrm{T}}\end{bmatrix}\\&=\begin{bmatrix}AH^T+Bh^t\\ CH^T+Dh^t\end{bmatrix}\end{aligned} \tag{12}$$

According to Schuer complement formula, obtain:

$$\begin{aligned}A&=\left(a^2I_L+H^TH-H^TH\left(a^2I_{L_0}+h^Th\right)h^TH\right)^{-1}\\B&=-AH^Th\left(a^2L_{L_0}+h^Th\right)^{-1}\\C&=-\left(a^2I_{L_0}+h^Th\right)^{-1}h^THA\\D&=-CH^{\mathrm{T}}h\left(a^2I_{L_0}+h^Th\right)^{-1}+\left(a^2I_{L_0}+h^Th\right)^{-1}\end{aligned} \tag{13}$$

Suppose $m = a^2 I_{L_0} + h^{\mathrm{T}}h - h^{\mathrm{T}}H\left(a^2 I_L + H^{\mathrm{T}}H\right)^{-1}H^{\mathrm{T}}h$. A can be obtained from the Sherman-Morrison-Woodbury formula.

$$\begin{aligned} A &= \left(a^2 I_L + H^{\mathrm{T}}H\right)^{-1} + \\ &\left(a^2 I_L + H^{\mathrm{T}}H\right)^{-1} H^{\mathrm{T}} h m^{-1} h^T H \left(a^2 I_L + H^{\mathrm{T}}H\right)^{-1} \end{aligned} \tag{14}$$

Eq. (15) can be obtained from Eq. (12)-(13).

$$\begin{aligned} AH^T + Bh^T &= \left(U^L + U^L h m^{-1} H U^L\right)\left(I - h\left(a^2 I_{L_0} + h^{\mathrm{T}}h\right)^{-1}h^T\right) \\ CH^T + Dh^T &= \left(k^2 I + h^T h\right)^{-1} h^T \left(I + H\left(AH^T + Bh^T\right)\right) \end{aligned} \tag{15}$$

IF-ELM algorithm gradually increases the hidden layer nodes in ELM algorithm, and the calculation results in each step are completely similar to the classical ELM algorithm, but the process of pseudo-inverse operation is eliminated, and the training efficiency is improved.

C. Inverse-free Extreme Learning Machine (IF-ELM)

In a previous study [18], Feng et al. introduced the Error Minimization Extreme Learning Machine (EM-ELM), which is a fast incremental learning algorithm. It allows for incremental updates of the output weights as the network grows. Experimental results demonstrated that EM-ELM outperforms other sequential or incremental algorithms in terms of generalization performance and speed. However, the EM-ELM algorithm has some drawbacks. Firstly, there might be rank deficit in the initial hidden layer's output matrix, leading to reduced calculation accuracy. Secondly, the algorithm's generalization performance is sometimes compromised by over-fitting. To tackle these concerns, a refined iteration of EM-ELM, known as IR-ELM, is introduced.

The central concept behind IR-ELM is to achieve swift computation of output weights while preserving the algorithm's ability to generalize. In the initial phase of R-ELM, a starting network comprising *N* randomly assigned hidden nodes is established, and subsequently, the output weights are computed utilizing the R-ELM algorithm. As new hidden nodes are gradually incorporated into the existing network, the output weights are recursively adjusted until the test accuracy reaches the desired criteria.

Assuming that there are already $S-1$ hidden nodes in the network, the output weight can be obtained, where H_{S-1} is the output matrix of the hidden layer, and subscript $S-1$ is the number of hidden nodes. When a new node is added, the output weight is $\beta_S = \left(C\mathrm{I} + \mathrm{H}_S^T \mathrm{H}_S\right)^{-1}\mathrm{H}_S^T\mathrm{T}$, where $\mathrm{H}_S = [\mathrm{H}_{S-1}, \mathrm{v}_S]$ is the hidden layer output matrix of the new network, and $\mathrm{v}_S = [g(W_S \cdot \mathrm{X}_1 + b_S), ..., g(W_S \cdot \mathrm{X}_N + b_S)]^T$ is the hidden layer output vector associated with the hidden node. Suppose $D_{S-1} = \left(C\mathrm{I} + \mathrm{H}_{S-1}^T H_{S-1}\right)^{-1}\mathrm{H}_{S-1}^T$ and $D_S = \left(C\mathrm{I} + \mathrm{H}_S^T \mathrm{H}_S\right)^{-1}\mathrm{H}_S^T$, and the left side of D_S can be written as:

$$\begin{aligned} \left(C\mathrm{I} + \mathrm{H}_S^T \mathrm{H}_S\right)^{-1} &= \left(C\mathrm{I} + \begin{bmatrix} \mathrm{H}_{S-1}^T \\ v_S^T \end{bmatrix}\left[\mathrm{H}_{S-1}, \mathrm{v}_S\right]\right)^{-1} \\ &= \begin{bmatrix} \mathrm{H}_{S-1}^T \mathrm{H}_{S-1} & \mathrm{H}_{S-1}^T \mathrm{v}_s \\ \mathrm{v}_S^T \mathrm{H}_{S-1} & v_S^T \mathrm{v}_s + C \end{bmatrix}^{-1} \end{aligned} \tag{16}$$

Since $C\mathrm{I} + \mathrm{H}_S^T \mathrm{H}_S$ is symmetric, the inverse matrix is also symmetric. So it can be expressed as:

$$\left(C\mathrm{I} + \mathrm{H}_S^T \mathrm{H}_S\right)^{-1} = \begin{bmatrix} \mathrm{A} & \mathrm{B} \\ B^T & \mathrm{E} \end{bmatrix}^{-1} = \begin{bmatrix} \mathrm{A}' & \mathrm{B}' \\ \mathrm{B}'^T & \mathrm{E}' \end{bmatrix} \tag{17}$$

where:

$$\begin{cases} \mathrm{A} = \mathrm{H}_{S-1}^T \mathrm{H}_{S-1} + C\mathrm{I} \\ \mathrm{B} = \mathrm{H}_{S-1}^T \mathrm{v}_S \\ \mathrm{E} = \mathrm{v}_S^T \mathrm{v}_S + C \end{cases} \tag{18}$$

So:

$$\mathrm{I} = \begin{bmatrix} \mathrm{A} & \mathrm{B} \\ B^T & \mathrm{E} \end{bmatrix}^{-1}\begin{bmatrix} \mathrm{A}' & \mathrm{B}' \\ \mathrm{B}'^T & \mathrm{E}' \end{bmatrix} = \begin{bmatrix} \mathrm{A}\mathrm{A}' + \mathrm{B}\mathrm{B}'^T & \mathrm{A}\mathrm{B}' + \mathrm{B}\mathrm{E}' \\ \mathrm{B}'^T\mathrm{A}' + \mathrm{E}\mathrm{X}'^T & \mathrm{B}^T\mathrm{B}' + \mathrm{E}\mathrm{E}' \end{bmatrix} \tag{19}$$

Find the solution to A', B', E':

$$\begin{cases} \mathrm{A}' = \dfrac{\mathrm{A}^{-1}\mathrm{B}\left(\mathrm{A}^{-1}\mathrm{B}\right)^T}{\mathrm{E} - \mathrm{B}^T\left(\mathrm{A}^{-1} - \mathrm{B}\right)} + \mathrm{A}^{-1} \\ \mathrm{B}' = \dfrac{\mathrm{A}^{-1}\mathrm{B}}{\mathrm{B}^T\mathrm{A}^{-1}\mathrm{B} - \mathrm{E}} \\ \mathrm{E}' = \dfrac{1}{\mathrm{E} - \mathrm{B}^T\mathrm{A}^{-1}\mathrm{B}} \end{cases} \tag{20}$$

The following results can be obtained:

$$D_S = \begin{bmatrix} \mathrm{A}' & \mathrm{B}' \\ \mathrm{B}'^T & \mathrm{E}' \end{bmatrix} \cdot \begin{bmatrix} H_{s-1}^T \\ \mathrm{v}_s^T \end{bmatrix} = \begin{bmatrix} \mathrm{A}'\mathrm{H}_{s-1}^T + \mathrm{B}'\mathrm{v}_S^T \\ \mathrm{B}'^T\mathrm{H}_{s-1}^T + \mathrm{E}'\mathrm{v}_S^T \end{bmatrix} \tag{21}$$

It can be expressed as:

$$D_s = \begin{bmatrix} L \\ M \end{bmatrix} \tag{22}$$

By substituting Eq. (18) and Eq. (19) into Eq. (22), obtain:

$$\begin{aligned} M &= \frac{\mathrm{B}^T\mathrm{A}^{-1}\mathrm{H}_{S-1}^T}{\mathrm{B}^T\mathrm{A}^{-1}\mathrm{B} - \mathrm{E}} + \frac{\mathrm{v}_S^{\mathrm{T}}}{\mathrm{E} - \mathrm{B}^T\mathrm{A}^{-1}\mathrm{B}} \\ &= \frac{\mathrm{v}_S^T\left(\mathrm{I} - \mathrm{H}_{S-1}D_{S-1}\right)}{\mathrm{v}_S^{\mathrm{T}}\left(\mathrm{I} - \mathrm{H}_{S-1}D_{S-1}\right)\mathrm{v}_S + C} \end{aligned} \tag{23}$$

$$\begin{aligned} L &= \frac{\mathrm{A}^{-1}\mathrm{B}\left(\mathrm{A}^{-1}\mathrm{B}\right)\mathrm{H}_{S-1}^T}{\mathrm{B}^T\mathrm{A}^{-1}\mathrm{B} - \mathrm{E}} + \frac{\mathrm{A}^{-1}\mathrm{B}\mathrm{v}_S^{\mathrm{T}}}{\mathrm{E} - \mathrm{B}^T\mathrm{A}^{-1}\mathrm{B}} + \mathrm{A}^{-1}\mathrm{H}_{S-1}^T \\ &= \frac{D_{S-1}\mathrm{v}_S\mathrm{v}_S^{\mathrm{T}}\left(\mathrm{H}_{S-1}D_{S-1} - \mathrm{I}\right)}{\mathrm{v}_S^{\mathrm{T}}\left(\mathrm{I} - \mathrm{H}_{S-1}D_{S-1}\right)\mathrm{v}_S + C} + D_{S-1} \\ &= D_{S-1}\left(\mathrm{I} - \mathrm{v}_S\mathrm{M}\right) \end{aligned} \tag{24}$$

The proposed IR-ELM algorithm can be summarized in Table III. Given a training dataset, highest number of hidden nodes allowed, and desired learning accuracy.

TABLE III. ALGORITHM PROCEDURE (IR-ELM)

Algorithm: IR-ELM

In relation to a given training data set $\psi = \{(\mathrm{X}_i, t_i) | \mathrm{X}_i \in R^n, t \in R^m, i = 1,\ldots,N\}$, the hidden node amount at the outset N_0, the largest hidden neuron number $N_{\max}$, and the speculated learning accuracy ε:

I Initialization neural network:

1. Randomize input weight vector W_i and basis $b_i, i = 1,\ldots,N_0$;

2. Evaluate the output matrix of hidden layer H_0;

3. Evaluate the output weight $\beta_0 : \beta_0 = D_0\mathrm{T} = \left(\mathrm{H}_0^T\mathrm{H}_0 + C\mathrm{I}\right)^{-1}\mathrm{H}_0^T\mathrm{T}$, where $\mathrm{T} = [t_1,\ldots,t_N]^{\mathrm{T}}$.

4. Set $s = 0$ and evaluate the learning accuracy $\varepsilon_s = \varepsilon_0$

II Recursive update network. When $N_S < N_{\max}$ and $\varepsilon_s > \varepsilon$;

1. 1. Set $s = s+1, N_s = N_{s-1}+1$;

2. Append a new randomly generated apply a hidden node to the existing network and calculate the corresponding output matrix $\mathrm{H}_s = [\mathrm{H}_{s-1}, \mathrm{v}_s]$;

3. The output weight is changed according to the following procedure:

$$\mathrm{M}_s = \frac{\mathrm{v}_s^T\left(\mathrm{I} - \mathrm{H}_{s-1}D_{s-1}\right)}{\mathrm{v}_s^T\left(\mathrm{I} - \mathrm{H}_{s-1}D_{s-1}\right)\mathrm{v}_s + C}$$

$$L_s = D_{s-1}\left(\mathrm{I} - \mathrm{v}_s\mathrm{M}_s\right)$$

$$\beta_s = D_s\mathrm{T} = \begin{bmatrix} L_s \\ \mathrm{M}_s \end{bmatrix}\mathrm{T}$$

4. Calculate the new learning precision ε_s.

IV. SIMULATION EXPERIMENT AND RESULT ANALYSIS

A. Performance Index

A soft-sensor model was developed utilizing the Incremental Extreme Learning Machine (I-ELM) neural network to predict the purity factor of the desired outcome compound in outlet N, the purity of impurities in outlet M, the yield of the target compound at outlet N, and the yield of impurities at outlet M in the SMB chromatography separation process. The ELM neural network model was configured with 30 hidden layer nodes.To train the soft-sensor model, the data set consisting of 1000 representative data points was compiled from historical data related to the simulated SMB chromatography separation process. Out of these, 900 data points were randomly selected as the training set, while the remaining 100 data points were reserved as the test set to evaluate the predictive performance of the model. Four specific indicators were selected shown in Table IV to evaluate the accuracy of the soft-sensor model's predictions. In these indicators, $\hat{y}$ is the estimated value and y is the actual value.

B. Forecasting Results and Analysis

The developed soft-sensing model for the SMB chromatographic separation process incorporates the subsequent assisting variables: the raw fluid inlet pump flow velocity (F pump), the pump flow performance for flushing fluid inlet (D pump), and the valve transition duration.The outputs of the soft-sensor models are the quality of the desired compound in the outflow at port N, the purity of impurities in the outflow at port M, the yield of the quality of the desired compound at port N, and the yield of impurities at port M. These outputs are predicted using the Incremental Extreme Learning Machine (IELM), Improved Fast Extreme Learning Machine (IFELM), Incremental Regularized Extreme Learning Machine (IRELM), and ELM.

The simulation results are presented in Fig. 4-11. In Fig. 4, the comparison of the Forecasting outputs for the purity level for the intended target substance in the effluent at port N is shown for IELM, IFELM, IRELM, and ELM. Fig. 5 displays the comparison of the Forecasting error curves. Fig. 6-7 demonstrate the predicted output comparison and Forecasting error curves for the impurity purity in the outlet R effluent. Similarly, Fig. 8-9 show the Forecasting outputs and error curves for the yield percentage for the target at port N. Finally, Fig. 10-11 present the Forecasting output and error curves for the yield of impurities at port M. Table V provides a comparison of the predictive performance indexes of the established soft-sensing models.

TABLE IV. PERFORMANCE INDEX OF SOFT-SENSOR MODEL

Name	Calculation method
MPE	$MPE = \max\left\{\left(\hat{y} - y\right), 0\right\}$
SSE	$SSE = \sum_{i=1}^{n}\left(\hat{y}_i - y_i\right)^2$
MAPE	$MAPE = \frac{\sum \frac{\left\lvert \hat{y}_i - y_i \right\rvert}{y_i} \times 100}{n}$
RMSE	$RMSE = \left[\frac{1}{n}\sum_{i=1}^{n}\left(\hat{y}_i - y_i\right)^2\right]^{\frac{1}{2}}$

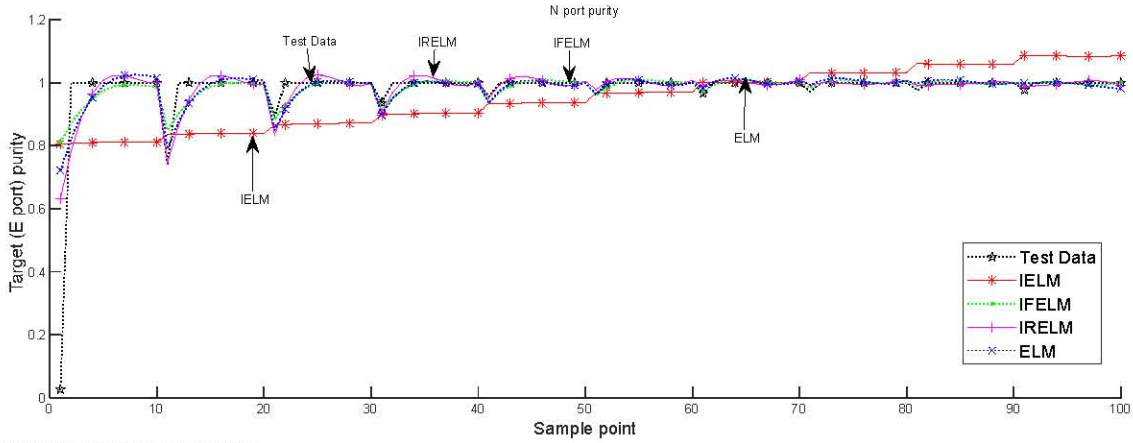

Fig. 4 Forecasting results of N port purity.

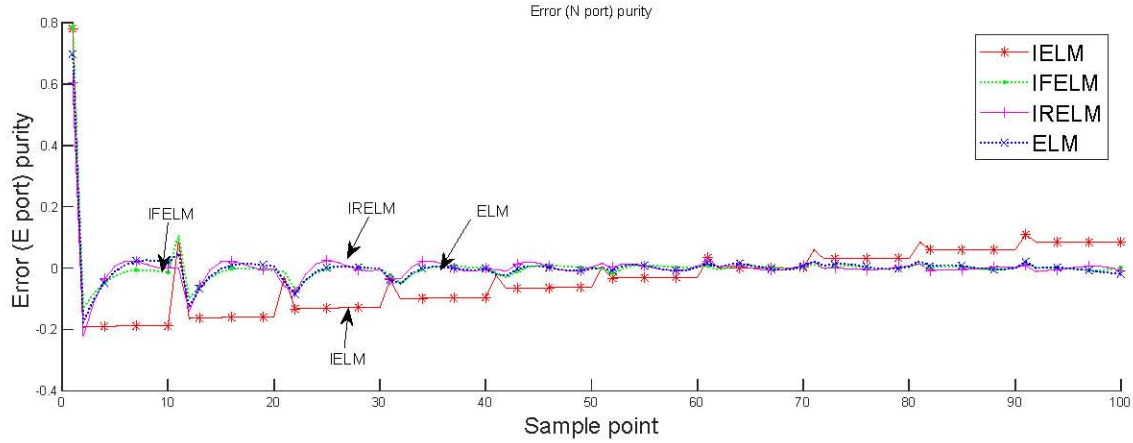

Fig. 5 Forecasting error of N port purity.

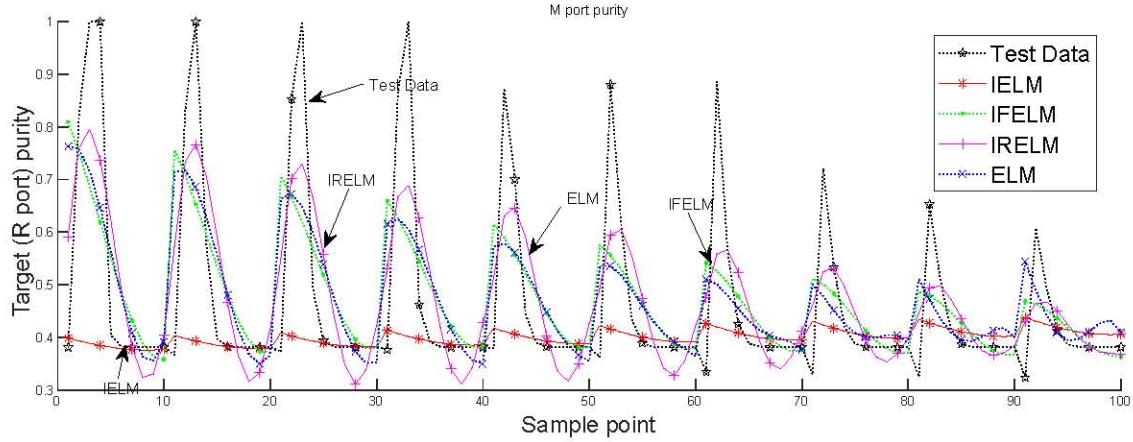

Fig. 6 Forecasting results of M port purity.

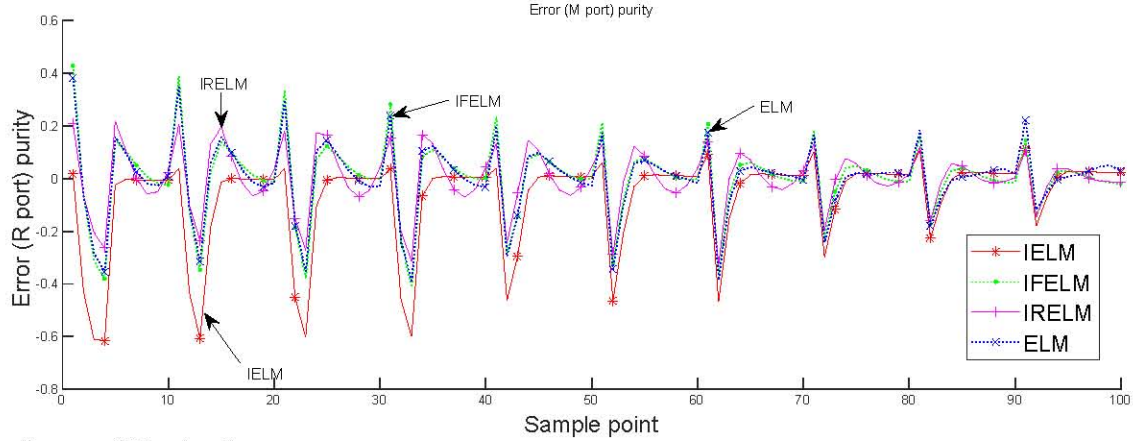

Fig. 7 Forecasting error of M port purity.

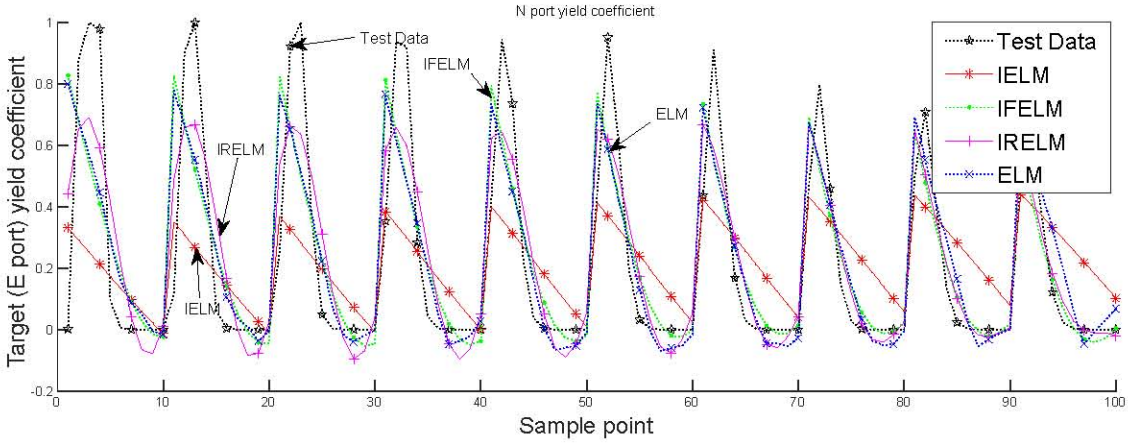

Fig. 8 Forecast result of N port yield.

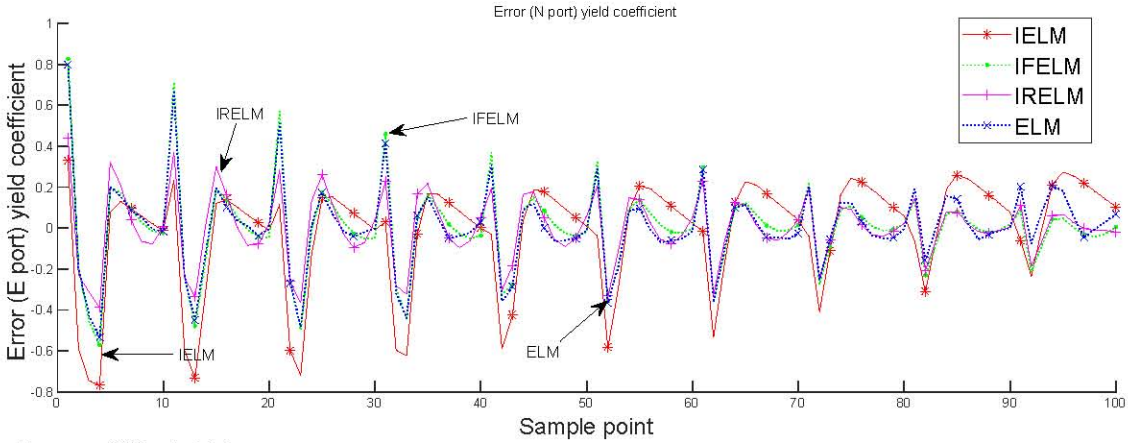

Fig. 9 Forecasting error of N port yield.

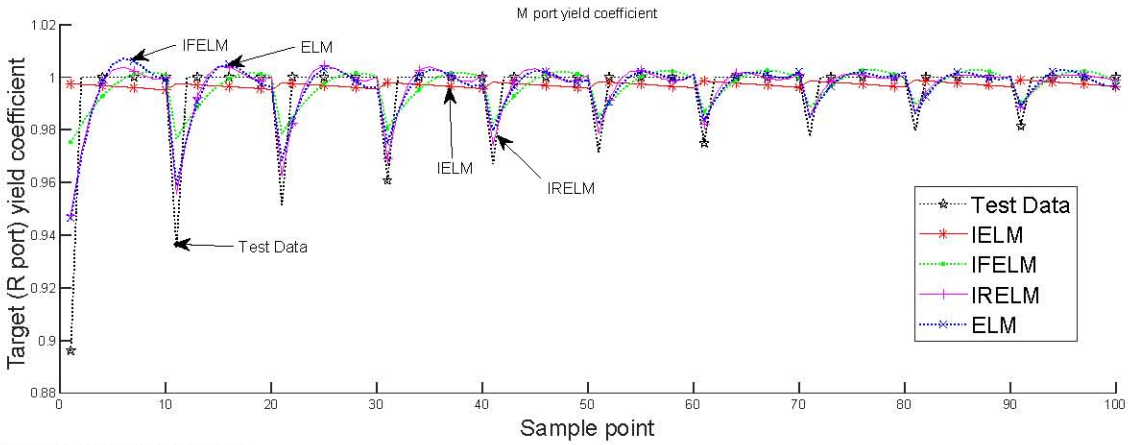

Fig. 10 Forecasting results of M port yield.

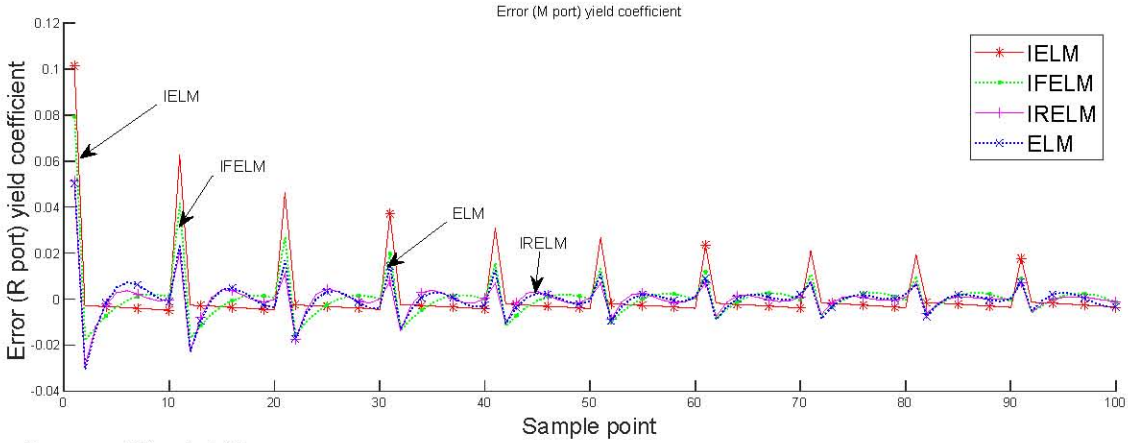

Fig. 11 Forecasting error of M port yield.

TABLE V. COMPARISON OF FORECASTING PERFORMANCE INDEXES OF SOFT-SENSOR MODELS

Performance index		RMSE	SSE	MAPE	MPE
N-port purity	IELM	0.1261	1.5897	0.9399	0.7791
	IFELM	0.0827	0.6843	0.0339	0.7850
	IRELM	0.0688	0.4730	0.0405	0.6049
	ELM	0.0758	0.5748	0.0406	0.6954
M-port purity	IELM	0.1933	3.7379	0.3950	0.6154
	IFELM	0.1473	2.1705	0.5363	0.4284
	IRELM	0.1167	1.3612	0.5277	0.3277
	ELM	0.1424	2.0284	0.5353	0.3943
N-port yield	IELM	0.2692	7.2482	0.2107	0.7662
	IFELM	0.2161	4.6717	0.3316	0.8257
	IRELM	0.1656	2.7427	0.2542	0.4400
	ELM	0.2087	4.3576	0.3009	0.7977
M-port yield	IELM	0.0148	0.0219	0.9871	0.1012
	IFELM	0.0111	0.0124	0.0069	0.0791
	IRELM	0.0079	0.0062	0.0065	0.0512
	ELM	0.0085	0.0072	0.0071	0.0503

V. CONCLUSION

A soft-sensor model was constructed for the SMB chromatographic separation process, considering the model inputs as the flow rate of the raw material liquid inlet pump (F pump), the flow rate of the flushing liquid inlet pump (D pump), and the valve switching time.The desired outputs of the model were determined to be the degree of contamination in the desired compound in the outflow liquid at port N, degree of impurity concentration in the outflow liquid at port M, the yield of the target compound at port N, and the yield of the impurity at port M.

By utilizing three incremental extreme learning machines, the soft-sensor model for the SMB chromatographic separation process was successfully established. The simulation results demonstrated a small error and a high level of Forecasting accuracy for degree of contamination in the desired compound in the outflow liquid at port N, the purity of impurities in the outflow liquid at port M, the yield of the target compound at port N, and the yield of impurities at port M. Moreover, the established soft-sensor model exhibited notable capability in accurately predicting the economic and technical indexes within the SMB chromatographic separation process.

REFERENCES

[1] Y. I. Lim, and S. B. Jorgensen, "A fast and accurate numerical method for solving simulated moving bed (SMB) chromatographic separation problems," *Chemical Engineering Science*, vol. 59, no. 10, pp. 1931-1947, 2004.

[2] S. F. Brown, M. D. Ogden, and E. S. Fraga, "Efficient simulation of chromatographic separation processes," *Computers & Chemical Engineering*, vol. 110, pp. 69-77, 2018.

[3] K. M. Kim, W. L. Ju, and S. Kim, "Advanced operating strategies to extend the applications of simulated moving bed chromatography," *Chemical Engineering & Technology*, vol. 40, no. 12, pp. 2163-2178, 2017.

[4] Z. Yan, J. S. Wang, S. Y. Wang, S. J. Li, D. Wang, and W. Z. Sun, "Model predictive control method of simulated moving bed chromatographic separation process based on subspace system identification," *Mathematical Problems in Engineering*, vol. 2019, pp. 1-24, 2019.

[5] S. Li, D. Wei, J. S. Wang, Z. Yan, and S. Y. Wang, "Predictive control method of simulated moving bed chromatographic separation process based on piecewise affine," *IAENG International Journal of Applied Mathematics*, vol. 50, no. 4, pp. 734-745, 2020.

[6] C. Xing, J. S. Wang, L. Zhang, and W. Xie, "Neural network soft-sensor modeling of PVC polymerization process based on data dimensionality reduction strategy," *Engineering Letters*, vol. 28, no. 3, pp. 762-776, 2020.

[7] Q. D. Yang, Y. Liu, J. S. Wang, Wang, S. H. Jiang, and X. L. Li, "Soft-sensing modeling of SMB chromatographic separation process based on ELM with variable excitation functions," *Engineering Letters*, vol. 30, no. 2, pp. 835-846, 2022.

[8] D. Wang, J. S. Wang, S. Y. Wang, and C. Xing, "Adaptive soft-sensor modeling of SMB chromatographic separation process based on dynamic fuzzy neural network and moving window strategy," *Journal of Chemical Engineering of Japan*, vol. 54, no. 12, pp. 657-671, 2021.

[9] D. Wang, J. S. Wang, S. Y. Wang, C. Xing, and X. D. Li, "ANFIS soft sensing model of SMB chromatographic separation process based on new adaptive population evolution particle swarm optimization algorithm," *Journal of Intelligent & Fuzzy Systems*, vol. 41, no. 6, pp. 6755-6780, 2021.

[10] D. Wang, J. S. Wang, S. Y. Wang, S. J. Li, Z. Yan, and W. Z. Sun, "Soft sensing modeling of the SMB chromatographic separation process based on the adaptive neural fuzzy inference system," *Journal of Sensors*, vol. 2019, pp. 1-16, 2019.

[11] G. B. Huang, Q. Y. Zhu, and C. K. Siew, "Extreme learning machine: theory and applications," *Neurocomputing*, vol. 70, no. 1-3, pp. 489-501, 2006.

[12] Y. Chen, X. Xie, T. Zhang, J. Bai, and M. Hou, "A deep residual compensation extreme learning machine and applications," *Journal of Forecasting*, vol. 39, no. 6, pp. 986-999, 2020.

[13] J. Zhao, Z. Zhou, and F. Cao, "Human face recognition based on ensemble of polyharmonic extreme learning machine," *Neural Computing & Applications*, vol. 24, no. 6, pp. 1317-1326, 2014.

[14] F. Lu, J. Wu, and J. Huang, "Restricted-boltzmann-based extreme learning machine for gas path fault diagnosis of turbofan engine," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 2, pp. 959-968, 2020.

[15] W. Xie, J. S. Wang, C. Xing, S. S. Guo, M. W. Guo, and L. F. Zhu, "Extreme learning machine soft-sensor model with different activation functions on grinding process optimized by improved black hole algorithm," *IEEE Access*, vol. 8, pp. 25084-25110, 2020.

[16] Q. An, R. Tang, and H. Su, "Robust configuration and intelligent MPPT control for building integrated photovoltaic system based on extreme learning machine," *Journal of Intelligent and Fuzzy Systems*, vol. 40, no. 17, pp. 1-18, 2021.

[17] M. S. G. García, E. Balsa-Canto, and A. V. Wouwer, "Optimal control of the simulated moving bed (SMB) chromatographic separation process," *IFAC Proceedings Volumes*, vol. 40, no. 5, pp. 183-188, 2007.

[18] G. Feng, G. B. Huang, Q. Lin, and R. Gay, "Error minimized extreme learning machine with growth of hidden nodes and incremental learning," *IEEE Transactions on Neural Networks*, vol. 20, no. 8, pp. 1352-1357, 2009.