Extreme Learning Machine Soft-sensing Model of SMB Chromatographic Separation Process Based on Multi-strategy Fusion Coati Optimization Algorithm

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

Abstract—**Simulated Moving Bed (SMB) chromatographic separation technology is an innovative method that combines traditional fixed-bed adsorption operation with real moving bed (TMB) chromatographic separation technology. In order to accurately evaluate the purity and yield of extracted and residual in the SMB chromatography separation process, a extreme learning machine soft-sensing model based on multi-strategy fusion coati optimization algorithm (COA) was proposed. This model selects auxiliary variables based on the analysis of the SMB chromatographic separation process. The proposed multi-strategy fusion COA includes logistic mapping, prey position vector and meme grouping to enhance the randomness and diversity of COA. Simulation results demonstrate that the optimized ELM can proficiently predict the key pecuniary and technical index of the SMB chromatography separation process.**

Index Terms—**SMB chromatographic separation process, Soft-sensor modeling, Extreme learning machine, Coati optimization algorithm, Multi-strategy fusion**

I. Introduction

SMB chromatography separation technology is a groundbreaking method of separation that has emerged from traditional fixed-bed adsorption operations and moving bed chromatography [1]. By periodically operating multiple columns to simulate the counter-current flow between two phases, this technology allows for a continuous supply of feed and removal of products. SMB chromatographic separation technology is widely recognized for its exceptional continuity, low energy consumption and high efficiency in adsorption separation processes. As a result, it has found extensive applications in various industries, including chemical, biological and food [2]. The SMB system exhibits non linearity, non-ideal characteristics, and non-equilibrium behavior in continuous production. With multiple degrees of freedom, this complex system faces challenges in simultaneously achieving optimal performance indicators, such as product purity, yield and mobile phase consumption in a stable periodic state. The separation mechanism of SMB chromatography is intricate, with various elements influencing the effectiveness of the separation process. The sensitivity of these operating elements and disturbances adds to the complexity of maintaining the process at its optimal operating point over an extended duration [3]. Obtaining accurate measurements of purity and yield of components presents considerable difficulties in real-world applications. Constraints, such as limited access to advanced detection devices and site limitations, hinder the ability to obtain real-time pecuniary and technical index of SBM chromatographic separation during actual production. Consequently, achieving direct quality closed-loop control becomes a significant challenge in practice [4-5]. The soft-sensor technology offers an effective solution by enabling the prediction of key indicators in complex industrial processes [6]. Ref. [7] investigated various auxiliary variables and key pecuniary index for soft-sensing models in SMB chromatographic separation technology, which can accurately predict the purity of the extract and raffinate solutions. A accommodating soft-sensing modeling method by utilizing the neuro-fuzzy network with dynamic structure and a temporal sliding window approach. Additionally, a soft-sensing model was implemented by combining the neuro-fuzzy network with with Kalman filter algorithm, linear least squares method and extended Kalman filter method [8]. Ref. [9] proposed a method for predicting the purity of extracts and raffinates in the SMB chromatography separation process by combining the improved particle swarm optimization algorithm with the least mean square method to optimize the neural fuzzy inference system. Ref. [10] proposed a soft-sensor modeling method based on the neural fuzzy system for predicting the component purity of extracts and raffinates in the SMB chromatographic separation process.

Manuscript received August 5, 2024; revised December 12, 2024. This work was supported by the Basic Scientific Research Project of Institution of Higher Learning of Liaoning Province (Grant No. LJ222410146054), and Postgraduate Education Reform Project of Liaoning Province (Grant No. LNYJG2022137).

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 (e-mail: xingcheng0811@163.com).

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).

Jie-Sheng Wang is a professor of School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, 114051, P. R. China (Corresponding author, phone: 86-0412-2538246; fax: 86-0412-2538244; e-mail: wang_jiesheng@126.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).

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).

Extreme Learning Machine (ELM) is a new type of feed-forward artificial neural network (ANN), which has a similar structure with the traditional back-propagation (BP) ANN, but with a fixed quantity of layers including the input layer, hidden layer and output layer. Unlike the BP ANN, the ELM model randomly assigns weights and offsets between the input layer and hidden layer. On the other hand, the weights between the hidden layer and output layer are determined through the least squares method, which minimizes the difference between the actual and desired output values. Thanks to its efficient training time, sweeping generalization capability and efficient performance, the ELM model has found widespread application in various fields, such as face recognition, fault diagnosis, soft-sensing modeling and intelligent control of photovoltaic systems. Consequently, it has proven to be an effective tool in these applications [11-16]. Additionally, ELM can handle large-scale datasets, which is crucial for soft-sensing as monitoring and estimation systems often involve vast amounts of sensor data. To address the issue of excessive hidden layer neurons in ELM, an improved version of ELM was proposed to provide an effective solution for its application in soft-sensing modeling technology [17]. The learning algorithm of the feed-forward ANN is an advanced technology that offers significant advantages. It greatly improves training efficiency and demonstrates strong generalization capabilities, making it highly practical in various real-world applications. One popular implementation of this algorithm is the extreme learning machine (ELM), which has gained widespread adoption in computer vision, natural language processing, time series analysis, and other fields [18-22]. However, using a fixed quantity of the hidden layer nodes in ELM may lead to a decline in prediction accuracy and stability due to the stochastic nature of the hidden layer biases and input weights. To tackle this issue, researchers have proposed various ELM variants that utilize swarm intelligence optimization algorithms, such as cuckoo search (CS), particle swarm optimization (PSO), grey wolf optimizer (GWO) and genetic algorithm (GA) [23-25]. In Ref. [26], the PSO algorithm was used to enhance the performance of ELM. To improve the accuracy of PV power generation prediction, a short-term PV power prediction model was presented by combining the variation mode decomposition, the improved archery algorithm and the improved ELM. Firstly, the PV data is decomposed into variation modes, and then a hybrid kernel function is applied to enhance the ELM. The archery algorithm is subsequently enhanced by using a random reverse learning strategy, followed by the utilization of the improved archery algorithm to optimize the nucleus elements of the hybrid nucleus-based ELM and establish a prediction model [27]. To derive the distribution law and prediction model for extreme rainfall in Hebei Province, researchers optimized the ELM model by using an improved pigeon swarm algorithm (MPIO-ELM), which was employed to forecast the extreme rainfall [28]. For addressing the issues of low accuracy and efficiency in predicting soil fertilizer supply in traditional agricultural irrigation systems, a crop soil fertilizer supply prediction model was introduced based on an improved sparrow search algorithm and ELM [29]. Additionally, Ref. [30] presents an integrated learning approach that enhances the ELM through the utilization of sample entropy and an improved Pathfinder algorithm. Accurate load forecasting greatly impacts the security, stability, and pecuniary benefit of power grids, thus making it a crucial component in power grid dispatching. To tackle the volatility caused by randomly generated input layer weights and hidden layer thresholds in ELM, Ref. [31] proposed a load prediction method that employs an improved GA to optimize ELM. Certain critical variables in the wastewater treatment process, like biochemical oxygen demand and chemical oxygen demand, can be challenging to measure accurately and promptly through traditional means. In order to overcome this challenge, a novel soft-sensing method is proposed based on ant-lion optimizer to construct a soft-sensing model [32].

Based on multi-strategy fusion COA, an ELM soft-sensing model of SMB chromatographic separation process was established with the component purity and yield of extract and residual solution as prediction index. The structure of the paper is described as follows. The second section introduces SMB chromatographic separation technology and soft-sensing model structure. The third section introduces the ELM. The fourth section introduces the COA and the multi-strategy fusion COA. The fifth section carries out the experimental simulation and result analysis. Finally, the conclusion of the paper is given.

II. SMB Chromatographic Separation Technology and Soft-sensor Modeling

A. SMB Chromatography Separation Technology

Simulating the reverse flow of the fixed phase adsorbent, the SMB chromatography continuously reverses the position of each inlet and outlet. The intention of SMB is to create a loop by connecting the starting and ending phases of multiple chromatographic columns. By systematically moving the positions of the eluent inlet, extraction outlet, raw material inlet and raffent outlet along the direction of the mobile phase, the flow of the fixed phase and the mobile phase simulate a counter-current movement. This setup allows for the separation of two components. In Fig. 1, the process principle of SMB chromatographic separation is illustrated by using the separation of components A and B as an example [27].

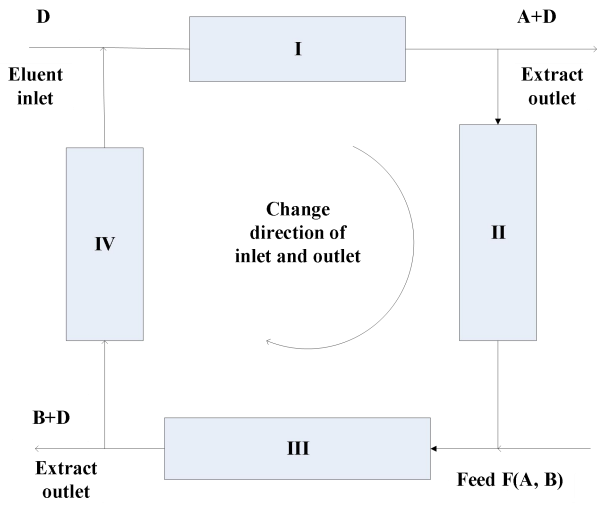

Fig. 1 Working principle of SMB chromatographic separation process.

Component A exhibits a stronger adsorption capacity than component B. The eluent used for desorption is represented by D, while N, F and M stand for the extract, feed and raffinate, respectively. The process can be divided into four zones, labeled as 1, 2, 3 and 4, each performing a specific function based on the position of the liquid inlet and outlet and the role of the entire bed.

B. Soft-sensor Model of SMB Chromatographic Separation Process

Measuring the purity and yield of product components in real time during the SMB chromatography adsorption separation process presents a significant challenge. Additionally, there are various factors that can affect the changes in component purity and yield. Therefore, the research on the soft-sensing model that can predict component purity and yield during the separation process holds great theoretical significance and engineering value. The fundamental concept behind soft-sensing technology is to use easily measurable variables as substitutes for complex or temporarily unavailable variables and then make estimations or inferences based on mathematical relationships. The soft-sensing model can be expressed as follows:

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

where, $\hat{X}$ is the estimated variable, d is the disturbance factor, u is the control input variable, y is the measurable output variable and X^* is the offline sampling value of the estimated variable or the analytical calculation value. The soft-sensing model structure of the SMB chromatography separation process is shown in Fig. 2.

Based on prior knowledge and the SMB chromatography separation process, the following variables have been selected as auxiliary variables in the model.

1) Flow rate of the feed-stock liquid at the inlet pump (F pump), measured in ml/min.

2) Flow rate of the flushing pump (D pump), measured in ml/min.

3) Time required for valve switching, measured in minutes.

Choose the following variables as output variables for the model:

1) Concentration of the desired substance in the effluent at port N. If there are impurities present at port N, the concentration will be less than 1.

2) Concentration of impurities in the effluent at port M. If there is a flow of the desired substance exiting at port M, the concentration will be less than 1.

3) Ratio of the mass of the desired substance flowing out at port N to the mass of the sampled desired substance, representing the yield of the desired substance at port N.

4) Ratio of the quality of the impurity flowing out at port M to the quality of the injected impurity, indicating the yield of impurity at port M.

The primary and auxiliary variables utilized in the SMB chromatography separation soft-sensor model can be found in Table I. The auxiliary variables act as inputs, while the purity of the desired substance in the effluent at port N, the purity of the impurity in the effluent at port M, the yield of the desired substance at port N and the yield of the impurity at port M are utilized as outputs. The interrelationship between these variables exhibits non-linearity and is modeled by using ELM, which helps develop a prediction model for relevant pecuniary and technical index. The soft-sensing model involves 1000 sets of data presented in Table II.

III. Extreme Learning Machine

The structure of ELM, a novel version of Single-hidden layer feed-forward neural network (SLFN), bears resemblance to the structure of BP ANN. To illustrate, Fig. 3 presents a diagram of the ANN model for ELM. consisting of three layers (the input layer, the hidden layer and the output layer). If the ELM model has n input variables x, then there will be n corresponding neurons in the input layer.

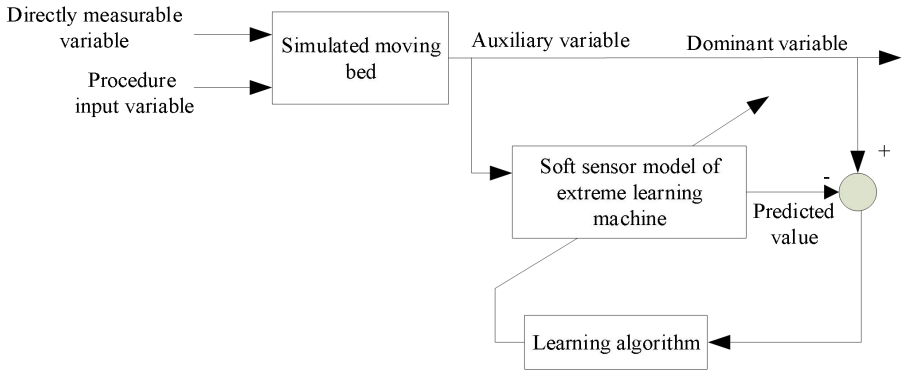

Fig. 2 Soft-sensor model structure.

Table I. Variable units and ranges

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 quantity	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

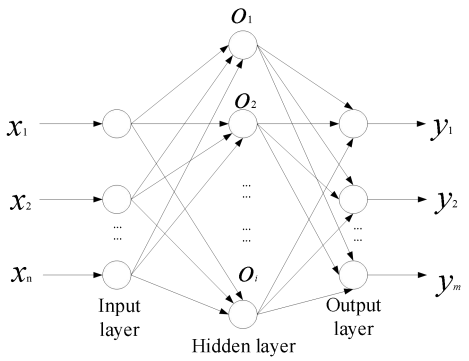

Fig. 3 ELM neural network structure.

Continue setting according to the corresponding rules, if the quantity of neurons in the prediction layer is m, then the quantity of output variable y is also m. If the weight of both the input layer and the hidden layer is w, then there is:

$$w=\begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1n} \\ w_{21} & w_{22} & \cdots & w_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{l1} & w_{l2} & \cdots & w_{\ln} \end{bmatrix}_{l\times n} \tag{2}$$

The fundamental structure of ELM is the fully connected structure. β is set to the link weight between the output layer and the hidden layer:

$$\beta=\begin{bmatrix} \beta_{11} & \beta_{12} & \cdots & \beta_{1m} \\ \beta_{21} & \beta_{22} & \cdots & \beta_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{l1} & \beta_{l2} & \cdots & \beta_{lm} \end{bmatrix}_{l\times m} \tag{3}$$

where, β_{ik} represents a connection weight of the kth neuron of the output layer and the jth neuron of the hidden layer. Set b as the threshold for hidden layer neurons:

$$b=\begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_l \end{bmatrix}_{l\times 1} \tag{4}$$

Continue with a training set, and then set its sample quantity to Q, and obtain two essential matrices, which are the output matrix Y and the input matrix X.

$$\begin{aligned} X&=\begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1Q} \\ x_{21} & x_{22} & \cdots & x_{2Q} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nQ} \end{bmatrix}_{n\times Q} \\ Y&=\begin{bmatrix} y_{11} & y_{12} & \cdots & y_{1Q} \\ y_{21} & y_{22} & \cdots & y_{2Q} \\ \vdots & \vdots & \ddots & \vdots \\ y_{m1} & y_{m2} & \cdots & y_{mQ} \end{bmatrix}_{m\times Q} \end{aligned} \tag{5}$$

Finally, $g(x)$ is set as the activation function of neurons in the hidden layer, which can be obtained from the ELM shown in Fig. 3, and the output T of ELM can be represented as:

$$\begin{aligned} T&=\begin{bmatrix} t_1 & t_2 & \cdots & t_Q \end{bmatrix}_{m\times Q} \\ t_j&=\begin{bmatrix} t_{1j} \\ t_{2j} \\ \vdots \\ t_{mj} \end{bmatrix}_{m\times 1}=\begin{bmatrix} \sum_{i=1}^{l}\beta_{i1}g\left(w_i x_j+b_i\right) \\ \sum_{i=1}^{l}\beta_{i2}g\left(w_i x_j+b_i\right) \\ \vdots \\ \sum_{i=1}^{l}\beta_{im}g\left(w_i x_j+b_i\right) \end{bmatrix}_{m\times 1} \quad j=1,2,\cdots,Q \end{aligned} \tag{6}$$

where, $w_i=\begin{bmatrix} w_{i1} & w_{i2} & \cdots & w_{in} \end{bmatrix}$, $x_i=\begin{bmatrix} x_{i1} & x_{i2} & \cdots & x_{in} \end{bmatrix}^T$. In this case, Eq. (6) can be expressed as:

$$H\beta=T' \tag{7}$$

where, T' is the transpose of T. H is the hidden layer output matrix of ELM, whose expression can be expressed as:

$$H\begin{pmatrix} w_1 & w_2 & \cdots & w_l & b_1 & b_2 & \cdots & b_l & x_1 & x_2 & \cdots & x_Q \end{pmatrix} = \begin{bmatrix} g\left(w_1x_1+b_1\right)\left(w_2x_2+b_2\right)\left(w_lx_1+b_l\right) \\ g\left(w_1x_2+b_1\right)\left(w_2x_2+b_2\right)\left(w_lx_2+b_l\right) \\ \vdots \\ g\left(w_1x_Q+b_1\right)\left(w_2x_Q+b_2\right)\left(w_lx_Q+b_l\right) \end{bmatrix}_{Q\times l} \tag{8}$$

If the quantity of training set samples in ELM is the same as the quantity of neurons in the hidden layer, then according to the w and b in the above formula, the ELM can also approximate the training samples with high precision, that is:

$$\sum_{j=1}^{Q}\left\|t_j-y_j\right\|=0 \tag{9}$$

where, $y_i=\begin{bmatrix} y_{1i} & y_{2i} & \cdots & y_{mi} \end{bmatrix}^t\ (j=1,2,\cdots Q)$. The actual training set sample Q is usually a relatively large value. In general, considering the computational efficiency, the quantity of hidden layer neurons K in the ELM ANN will not be more than the amount of Q. The training error of the ELM model can approximate an arbitrary $\varepsilon>0$, namely:

$$\sum_{j=1}^{Q}\left\|t_j-y_j\right\|<\varepsilon \tag{10}$$

If the activation function $g(x)$ of the ELM is continuously differentiable, the elements in the ELM model can experience minimal changes. Both w and b remain constant throughout the training process, and their initial values are arbitrarily initialized. During this operation, the solutions of the subsequent equations can be considered as specific values, representing the connection weights between the hidden layer and the output layer of ELM.

$$\min_{j=1}(H\beta-T^{'}) \tag{11}$$

The calculation result of Eq.(11) is as follows:

$$\hat{\beta}=H^{+}T^{'} \tag{12}$$

where, H^{+} is called the Moore-Penrose generalized inverse of the output matrix H of the hidden layer of ELM. It should be noted here that it takes a lot of learning time to compute the inverse matrix efficiently, so the solution of the inverse matrix is very important.

It can be seen from the above steps that the weight w and offset b of ELM are all arbitrary choices, and how to deal with these two values will greatly affect the results. Therefore, it is important not only to know the excitation function $g(x)$ and its element β of the hidden layer neurons, but also consider the quantity of hidden layer neurons in ELM. The following is the specific training process of ELM.

(1) Firstly, the quantity of neurons is given, and the weight w and offset b of the ELM model are arbitrarily selected;

(2) Given the qualified activation function of ELM, H in the ELM model is calculated;

(3) Finally, the output layer weight $\hat{\beta}:\hat{\beta}=H^{+}T^{'}$ in the ELM model is calculated.

IV. Coatis Optimization Algorithm

A. Algorithm Initialization Process

The natural behaviors of coatis have served as the foundational inspiration for the design of COA. In the COA, Coatis are considered as members of the population, and their position in the search space determines the value of the optimal variable. Therefore, the placement of Coatis in COA represents a potential solution to the problem being addressed. The COA implementation starts by randomly assigning the positions of Coatis in the search space by using Eq. (13).

$$\begin{aligned} &X_i:\chi_{ij}=lb_j+r\left(ub_j-lb_j\right), \\ &i=1,2,...,N,\ j=1,2,...,m \end{aligned} \tag{13}$$

where, X_i is the position of the ith Coati in the search space, $X_{i.i}$ is the value of the jth optimal variable, N is the quantity of Coatis, m is the quantity of optimal variables, r is a random real quantity between 0 and 1, and lb_i and ub_i are the local upper and lower limits of the jth optimal variables, respectively. The population matrix X represents the coatis population in COA mathematically.

$$X=\begin{bmatrix} X_1 \\ \vdots \\ X_i \\ \vdots \\ X_N \end{bmatrix}_{N\times m}=\begin{bmatrix} x_{1,1} & \cdots & x_{1,j} & \cdots & x_{1,m} \\ \vdots & \ddots & \vdots & \iddots & \vdots \\ x_{i,1} & \cdots & x_{i,j} & \cdots & x_{i,m} \\ \vdots & \iddots & \vdots & \ddots & \vdots \\ x_{N,1} & \cdots & x_{N,j} & \cdots & x_{N,m} \end{bmatrix}_{N\times m} \tag{14}$$

The intention function is evaluated with different values by associating optimal variables with candidate solutions. These values are expressed through Eq. (15).

$$F=\begin{bmatrix} F_1 \\ \vdots \\ F_i \\ \vdots \\ F_N \end{bmatrix}_{N\times 1}=\begin{bmatrix} F\left(X_1\right) \\ \vdots \\ F\left(X_i\right) \\ \vdots \\ F\left(X_N\right) \end{bmatrix}_{N\times 1} \tag{15}$$

where, F is the vector of the obtained intention function and F is the value of the intention function based on the ith Coati.

The COA incorporates the intention function value to evaluate the quality of candidate solutions. Consequently, the population member that produces the best evaluation of the intention function is identified as the best member of the population. Throughout the iterations of the algorithm, the candidate solutions undergo modifications, leading to updates in the best members of the population at each iteration.

B. Mathematical Model of COA

The process of updating the areas where Coatis are found in COA involves simulating two natural behaviors of the animals: attacking iguanas and avoiding predators. This results in two distinct phases of revitalization for the COA population.

(1) Phase 1. It is known as the exploration phase to mimic the hunting and attack strategies used by Coatis when attacking iguanas. Some Coatis climb trees to scare the iguanas down to the ground, while others wait under trees for them to fall. Once the iguanas are on the ground, the Coatis attack and capture them. This strategy allows COA to explore different locations within the searching space, and enabling global search capabilities and thorough exploration of the problem-solving space.

The design of COA assumes that the best member of the population is located where the iguana is. Another approach is to divide the Coatis into two groups: half of them climb trees and the other half wait for the iguanas to fall to the ground. This arrangement maximizes the collaborative abilities of the Coatis and increases the success rate of hunting iguanas. By connecting the location of the best member to that of the iguana and organizing the Coatis into two groups with different roles, COA effectively harnesses the population members' collaborative abilities to enhance problem solving and improve the overall search process.

The mathematical simulation for the position of the Coatis climbing the tree is represented by Eq. (16).

$$\begin{aligned} &X_i^{P1}: x_{i,j}^{P1} = x_{i,j} + r\cdot\left(\lg uana_j - I\cdot x_{i,j}\right), \\ &for \quad i = 1,2,\ldots,\left\lfloor \frac{N}{2} \right\rfloor \quad and \quad j = 1,2,\ldots,m \end{aligned} \tag{16}$$

After the iguana lands, it is randomly positioned within the search space. In accordance with this random placement, the coatis on the ground will navigate within the same search space. This navigation process is simulated by using Eq. (17)-(18).

$$\begin{aligned} &\lg uana^G: \lg uana_j^G = lb_j + r\cdot\left(ub_j - lb_j\right), \\ &j = 1,2,\ldots,m \end{aligned} \tag{17}$$

$$\begin{aligned} &X_i^{P1}: x_{i,j}^{P1} = \begin{cases} x_{i,j} + r\cdot\left(\lg uana_j^G - I\cdot x_{i,j}\right), F_{\lg uana^G} < F_i \\ x_{i,j} + r\cdot\left(x_{i,j} - \lg uana_j^G\right), else \end{cases} \\ &for \quad i = \left\lfloor \frac{N}{2} \right\rfloor + 1, \left\lfloor \frac{N}{2} \right\rfloor + 2,\ldots,N \quad and \quad j = 1,2,\ldots,m \end{aligned} \tag{18}$$

If the intention function value increases, the new position calculated for each Coati is considered acceptable during the update process. If the intention function value does not increase, the Coati will stay in its original position. This update condition is applicable to $i = 1,2,\ldots,N$ and is represented by Eq. (19).

$$X_i = \begin{cases} X_i^{P1}, F_i^{P1} < F_i \\ X_i, else \end{cases} \tag{19}$$

where, X_i^{P1} is the new position calculated for the *ith* coatis; $X_{i,j}^{P1}$ is its *jth* magnitude; F_i^{P1} is its intention function value; r is a random real quantity between 0 and 1; $\lg uana$ represents the iguana's position in the search space, actually referring to the member with the best position; $\lg uana_j$ is its *jth* magnitude; I is an integer, randomly selected from the set $\{1 \quad 2\}$; $\lg uana^G$ of the iguana's position on the ground is randomly generated, with $\lg uana_j^G$ as its *jth* magnitude; $F_{\lg uana^G}$ is the value of its intention function and $\lfloor \cdot \rfloor$ is the base function (also known as the largest integer function).

(2) Phase II: Predator Escape Development Phase. In this phase, the focus is on developing a mathematical model that represents the natural behavior of coatis when encountering and fleeing from predators. When a coatis is under attack, it quickly reacts to escape its current position. The COA aims to simulate the coatis' behavior, demonstrating effective exploitation during local searches. Its strategy involves relocating the coatis to a safer area while minimizing the distance from its current location. This showcases COA's ability to effectively use its own skills in local search, thereby enhancing problem-solving effectiveness. By mathematically modeling the coatis' predator evasion behavior, COA can leverage the information within the search space to obtain superior solutions. To simulate this behavior, random positions near the coatis' locations are generated by using Eq. (20)-(21).

$$lb_j^{local} = \frac{lb_j}{t}, ub_j^{local} = \frac{ub_j}{t}, where \quad t = 1,2,\ldots,T \tag{20}$$

$$\begin{aligned} &X_i^{P2}: x_{i,j}^{P2} = x_{i,j} + (1-2r)\cdot\left(lb_j^{local} + r\left(ub_j^{local} - lb_j^{local}\right)\right) \\ &for \quad i = 1,2,\ldots,N \quad j = 1,2,\ldots,m \end{aligned} \tag{21}$$

It is acceptable if the newly calculated position raises the value of the intention function, and this condition is simulated by using Eq. (22).

$$X_i = \begin{cases} X_i^{P2}, F_i^{P2} < F_i \\ X_i, else \end{cases} \tag{22}$$

where, X_i^{P2} is the new location of the calculation of the *ith* Coatis in the second stage of COA; $X_{i,j}^{P2}$ is its *jth* magnitude; F_i^{P2} is its intention function value; r is a random real quantity between 0 and 1; t is the quantity of iterations; lb_j^{local} and ub_j^{local} are the local upper and lower limits of the *jth* optimal variable, and lb_j and ub_j are the upper and lower limits of the *jth* optimal variable, respectively.

After updating the Coatis' locations in the search space through the first and second stages, the iteration of the COA comes to an end. The population renewal process is carried out according to Eq. (16)-(20). This process continues until the final iteration of the algorithm is reached. Upon completing a run of the COA, the best solution obtained during all iterations is returned as the output. The various stages of COA implementation are illustrated in the flow chart in Fig. 4.

C. Multi-strategy Fusion Coati Optimization Algorithm

(1) COA Based on Chaotic Mapping

In this section, the hybrid Coati Optimization Algorithm (HCOA) was proposed by incorporating an enhanced chaotic mapping technique. The key concept behind this approach is to utilize a chaotic map for generating a chaotic sequence, which is essentially a sequence of randomness derived from a simple deterministic system.

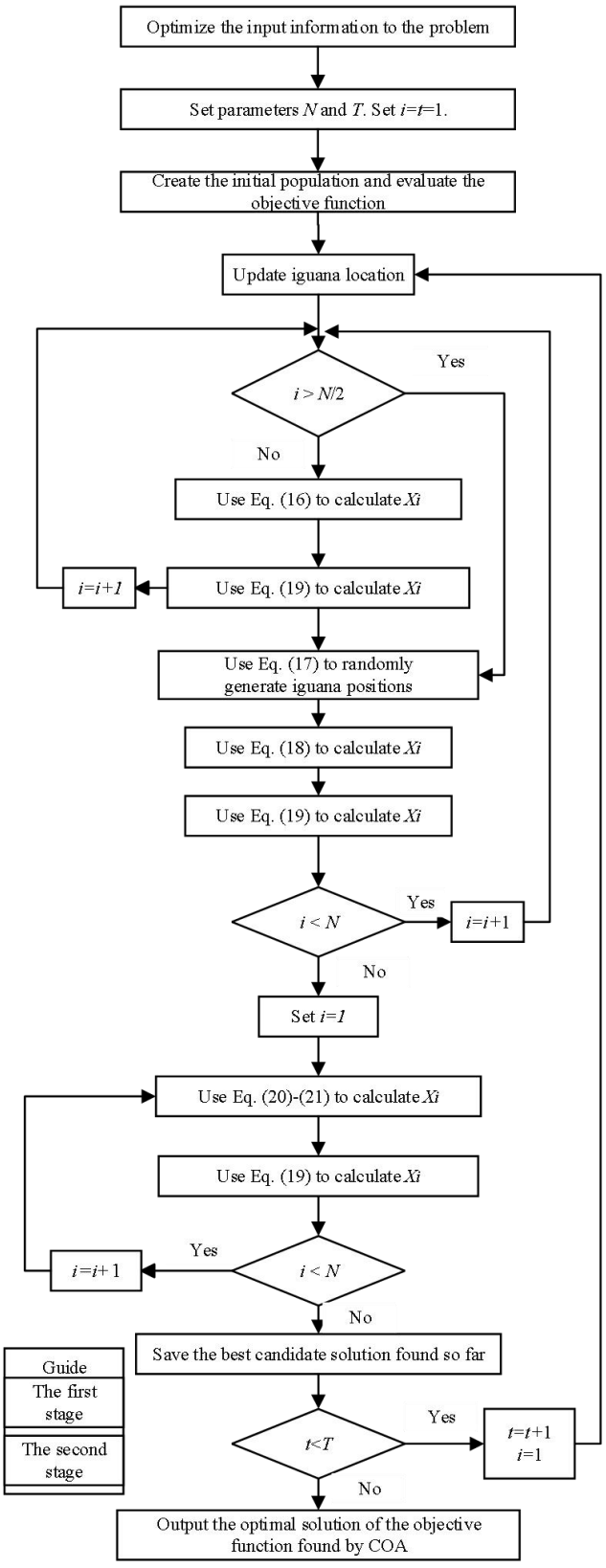

Fig. 4 Flowchart of COA.

In the realm of optimization, chaotic maps can serve as alternatives to pseudo-random quantity generators, enabling the generation of chaotic quantity within the range of 0 to 1. Experimental evidence has demonstrated that incorporating chaotic sequences in the initialization, selection, crossover and mutation stages of the population significantly influences the overall algorithmic process, oftentimes resulting in improved outcomes compared to the utilization of pseudo-random quantity. This study introduces the Logistic mapping depicted visually in Fig. 5 and represented by Eq. (23).

$$z_{k+1} = \mu z_k (z_k - 1)$$
$$z_0 \subset \{0, 0.25, 0.5, 0.75, 1.0\}, \mu \subset [0,4] \tag{23}$$

To enhance the level of randomness and diversity within the population, the COA incorporates a chaotic map during the initial stage of population initialization. In the original version of COA, the population is randomly initialized by generating a position matrix within predetermined upper and lower limits. However, with the integration of chaotic mapping, each individual in the population creates its position matrix by using Eq. (24). This integration allows for a more varied and diverse population, contributing to improved optimization outcomes.

$$\begin{cases} mt_i = rand & i = 1 \\ mt_i = u \times mt_{i-1} \times (mt_{i-1}) & i > 1 \end{cases} \tag{24}$$

(2) COA Based on Meme Grouping

The shuffled frog leaping algorithm (SFLA) is a recently developed heuristic population evolution algorithm that offers efficient computing performance and exceptional global search capability. It effectively combines the desirable traits and advantages of meme-based algorithms and PSO algorithm. This approach can be described as the fusion of meme algorithms and PSO algorithm to utilize their respective advantages to accomplish an efficient and precise optimization process on COA.

Suppose that the initial population formed by $L = F_1, F_2, ..., F_N$ is a group of frogs, where $F_i = x_{i1}, x_{i2}, ..., x_{iS}$ represents the ith frog in the S-magnitude space in the problem. In addition, each frog in this population is ranked in descending order according to fitness value. Let m be the quantity of memes that the whole population is divided into, the first frog enters the first meme group, the second frog enters the second meme group, the m frog enters the m meme group, the $m+1$ frog enters the $m+1$ meme group, and so on, until all the frogs are allocated. At the same time, the frog with the best fitness in each meme group is labeled as F_b, the frog with the worst fitness is labeled as F_w, and the frog with the best fitness in the whole population is labeled as F_g. Then, the local position update operations are performed for frogs in each meme group. The specific update formula is described as follows:

$$\begin{cases} D = r(F_b - F_w) \\ F_w = F_w + D \end{cases} (-D_{\max} \le D \le D_{\max}) \tag{25}$$

where, r is a quantity randomly generated from 0 to 1, the jump distance is represented by D, the current position of the frog is represented by F_w, and the maximum jump distance of the frog is represented by $D_{\max}$. Assuming that the updated position is better than before, the original updated frog F_w can be replaced with a new frog. Otherwise, replace F_b with F_g and perform the local location update operation.

$$\begin{cases} D = r\left(F_g - F_w\right) \left(-D_{\max} \le D \le D_{\max}\right) \\ F_w = F_w + D \end{cases} \tag{26}$$

If the frog doesn't find a better solution by executing Eq. (26), or if it exceeds the maximum distance allowed, it will be replaced by a randomly generated frog. This process of updating local locations is repeated multiple times, while the frogs in all the groups are rearranged and separated. The next set of local location updates is then carried out. The iterative process continues until a specific convergence condition is met or the maximum quantity of mixing iterations is reached. By repeatedly going through this loop, the frogs' positions are gradually optimized to better achieve the algorithm's intention.

The COA incorporates the concept of leapfrog meme grouping and it is named as SCOA. The idea behind it is as follows. In the initial stage of COA, half of the coatis climb the tree while the other half wait for the iguana to fall to the ground. This grouping method is randomized. By incorporating the grouping approach used in the SFLA, the COA sorts the population based on fitness from largest to smallest. The even-quantityed coatis then wait for the iguana to fall to the ground.

$$\begin{aligned} &X_i^{P1} : x_{i,j}^{P1} = x_{i,j} + r \cdot \left(\lg uana_j - I \cdot x_{i,j}\right), \\ &for \quad i = 1,3,5,... and \quad j = 1,2,\ldots,m \end{aligned} \tag{27}$$

(3) COA Based on Prey Position Vector

The gray wolf, a member of the canine family, is an apex predator and holds the highest position in the food chain. They typically live in packs consisting of 5 to 12 individuals. These packs have a fascinating social hierarchy, as shown in Fig. 6 [34]. The alpha wolf, who is the leader, makes decisions regarding hunting locations, sleeping arrangements and waking times. Assisting the alpha is the beta wolf, the second-ranking member in the pack. At the bottom of the hierarchy is the omega wolf, often targeted by more dominant wolves and serving as a scapegoat.

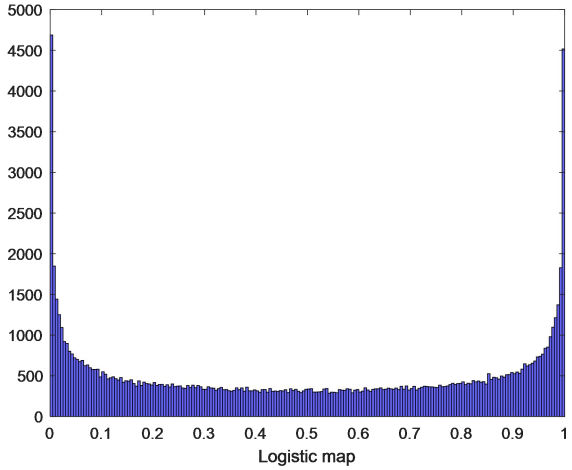

Fig. 5 Logistic mapping image.

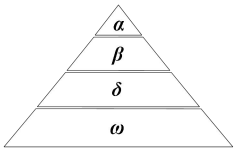

Fig. 6 Hierarchy of grey wolves.

In some cases, omegas also care for the young in the pack. Wolves without alpha, beta or omega status are considered subordinate wolves and are referred to as delta wolves. These wolves are obligated to follow the instructions of alphas and betas but ultimately fall under the authority of the omega wolves. The gray wolf demonstrates important social behaviors such as pack hunting and establishing a social hierarchy. The hunting process was categorized into several stages, including tracking, chasing and approaching the prey. This study utilized mathematical modeling to describe both the hunting technique and the social ranking within gray wolf packs. The research resulted in the development of the gray wolf optimizer, which has been applied to optimize various tasks, showcasing its practical application beyond the understanding of wolf behavior.

In this section, a coatis optimization algorithm with improved prey position vector (GCOA) is proposed. The vector coefficient of the position update formula in GWO is introduced to replace the random quantity r in the COA, and the vector coefficient $A = 2a * r_1 - a$ in the GWO is introduced, where a is the real quantity that linearly decreases from 2 to 0 in the iterative process, and r_1 is the random real quantity from 0 to 1. Substitute A into Eq. (26)-(27) to obtain:

$$\begin{aligned} &X_i^{P1} : x_{i,j}^{P1} = x_{i,j} + \mathrm{A} \cdot \left(\lg uana_j - I \cdot x_{i,j}\right), \\ &for \quad i = 1,2,\ldots,\left\lfloor \frac{N}{2} \right\rfloor and \quad j = 1,2,\ldots,m \end{aligned} \tag{28}$$

$$\begin{aligned} &X_i^{P1} : x_{i,j}^{P1} = \begin{cases} x_{i,j} + \mathrm{A} \cdot \left(\lg uana_j^G - I \cdot x_{i,j}\right), F_{\lg uana^G} \langle F_i \\ x_{i,j} + \mathrm{A} \cdot \left(x_{i,j} - \lg uana_j^G\right), else \end{cases} \\ &for \quad i = \left\lfloor \frac{N}{2} \right\rfloor + 1, \left\lfloor \frac{N}{2} \right\rfloor + 2, \ldots, N \quad and \quad j = 1,2,\ldots,m \end{aligned} \tag{29}$$

In addition, the position update formula in the exploration stage of the COA is an integer I, that is, randomly selected from the set $\{1,2\}$. In order to increase its accuracy, I is improved to $I_{NEW} = e^i$, which is the exponential multiple changing with the quantity of populations. Then it is substituted into the Eq. (29) to obtain:

$$\begin{aligned} &X_i^{P1} : x_{i,j}^{P1} = \begin{cases} x_{i,j} + \mathrm{A} \cdot \left(\lg uana_j^G - I_{NEW} \cdot x_{i,j}\right), F_{\lg uana^G} \langle F_i \\ x_{i,j} + \mathrm{A} \cdot \left(x_{i,j} - \lg uana_j^G\right), else \end{cases} \\ &for \quad i = \left\lfloor \frac{N}{2} \right\rfloor + 1, \left\lfloor \frac{N}{2} \right\rfloor + 2, \ldots, N \quad and \quad j = 1,2,\ldots,m \end{aligned} \tag{30}$$

(4) Flowchart of Multi-strategy COA

Based on the above three strategies to improve COA, a multi-strategy fusion coatis optimization algorithm (SHGCOA) is proposed by combining these three strategies. Its pseudo-code is shown in Table Ⅲ.

V. Simulation Experiment and Result Analysis

A. Performance Index

To facilitate the SMB chromatographic separation process, the soft-sensor models were established for various elements including the purity of the target at exit N, the purity of the impurity at exit M, the yield of the target at exit

N and the yield of the impurity at exit M. These models were built on an optimized ELM with a hidden layer consisting of 30 nodes. A set of 1000 representative data sets was selected from historical data related to SMB chromatography separation. Among these, 900 data sets were randomly assigned as training sets for the ELM model, while the remaining 100 were utilized as test sets to assess the predictive performance of the soft-sensor model. In order to effectively compare the prediction abilities of the soft-sensor model, four specific indicators were selected for evaluation, which are maximum percentage error (MPE), sum of squares error (SSE), mean absolute percentage error (MAPE) and root mean square error (RMSE). These indicators are presented in Table IV, where $\hat{y}$ is the estimated value and y is the actual value.

B. Soft-sensor Model of SMB Chromatographic Separation Process Based on COA with Single Strategy

To optimize the ELM model, a simulation experiment was conducted by using the enhanced COA algorithm with a single strategy. The established soft-sensor model for the SMB chromatographic separation process includes additional variables such as the flow rate of the feedstock liquid inlet pump (F pump), the flow rate of the flushing liquid inlet pump (D pump) and the valve switching time. This model provides outputs for the purity of the target in the effluent at port N, the purity of the impurity in the effluent at port M, as well as the yield of the target and impurity at their respective ports. The optimized ELM accurately captures the non-linearity between these variables, enabling the development of a prediction model for the corresponding pecuniary and technical indices.

Fig. 7-14 presents the simulation results of the SMB chromatographic separation process. In Fig. 7, the predicted output curves for different purities of the target in the effluent at exit N are compared. These purities include ELM, COA-ELM, HCOA-ELM, SCOA-ELM and GCOA-ELM. Fig. 8 displays the comparison of prediction error curves. Similarly, Fig. 9 shows the predicted output curves for the purity of impurities in the effluent at port M, while Fig. 10 illustrates the respective prediction error curves. Fig. 11 showcases the comparison curves of the prediction output for the yield of the target in the N-port under different purities, namely ELM, COA-ELM, HCOA-ELM, SCOA-ELM and GCOA-ELM, whereas Fig. 12 depicts the corresponding prediction error comparison curves. Furthermore, Fig. 13 demonstrates the comparison curves for the predicted output of the yield of impurities at port M by using ELM, COA-ELM, HCOA-ELM, SCOA-ELM and GCOA-ELM, while Fig. 14 presents the comparison curves for the prediction error. Lastly, Table V provides a comparison of the predictive performance indicators of the established soft-sensing models. By analyzing the simulation experiment chart, it becomes evident that the optimized versions of COA-ELM, HCOA-ELM, SCOA-ELM and GCOA-ELM provide more accurate predictions compared to the unoptimized ELM model for elements such as N-port purity, M-port purity, N-port yield and M-port yield. Additionally, the ELM model optimized by using the improved COA demonstrates superior performance and greater precision compared to the unimproved optimization models.

C. Soft-sensor Model of SMB Chromatographic Separation Process Based on Multi-strategy Fusion COA

SHGCOA-ELM algorithm is developed by combining the improved HCOA-ELM, SCOA-ELM and GCOA-ELM based on the results obtained from the previous experiments. Two other optimized models (SSA-ELM[35] and GGO-ELM[36]) are selected for simulation and comparison to evaluate the performance of the prediction models. The soft-sensor model for the SMB chromatographic separation process includes auxiliary variables such as F pump (flow rate of the feed-stock liquid inlet pump), D pump (flow rate of the flushing liquid inlet pump), and valve switching time. The soft-sensor model outputs consist of the purity of the target at port N, the purity of the impurity at port M, the yield of the target at port N and the yield of the impurity at port M. Utilizing the optimized ELM, a nonlinear relationship is established among these variables, enabling the creation of a prediction model for corresponding pecuniary and technical index. The simulation results are illustrated in Fig. 15-22. Table VI provides a comparison of the performance indicators of the established soft-sensing models.

TABLE III. PSEUDO-CODE OF SHGCOA

SHGCOA calculation process pseudo-code
Initialize the population size N and the maximum quantity of iterations T
Initialize the population and calculate the fitness value for each individual
For i=1:T
Determine the position of the optimal individual xbest, which is the position of the iguana on the tree
% Phase 1
Sort by population fitness
An individual with an odd quantity
Use Eq. (25) to update each individual position
Use Eq. (14) to make greedy choices
An individual with an even quantity
The position of each individual is updated using Eq. (17) and Eq. (30)
Use Eq. (14) to make greedy choices
% Phase 2
For i=1:N
The position of each individual is updated using Eq. (20)-(21)
Use Eq. (14) to make greedy choices
End
Store the optimal solution and the optimal value
End
Output the optimal value and the optimal solution

TABLE IV. PERFORMANCE INDEXES OF SOFT-SENSOR MODEL

Index	Function
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	$MAPE = \frac{\sum \left\lvert \frac{\hat{y}_i - y_i}{y_i} \right\rvert \times 100}{n}$
RMSE	$RMSE = \left[\frac{1}{n}\sum_{i=1}^{n}\left(\hat{y}_i - y_i\right)^2\right]^{\frac{1}{2}}$

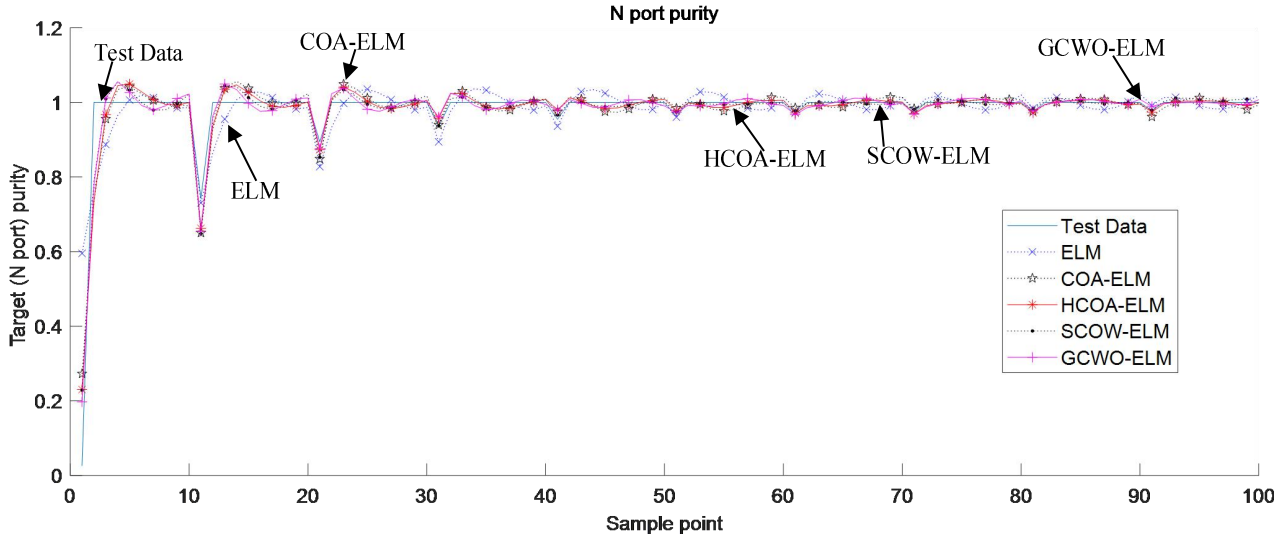

Fig. 7 Prediction results of N-port purity.

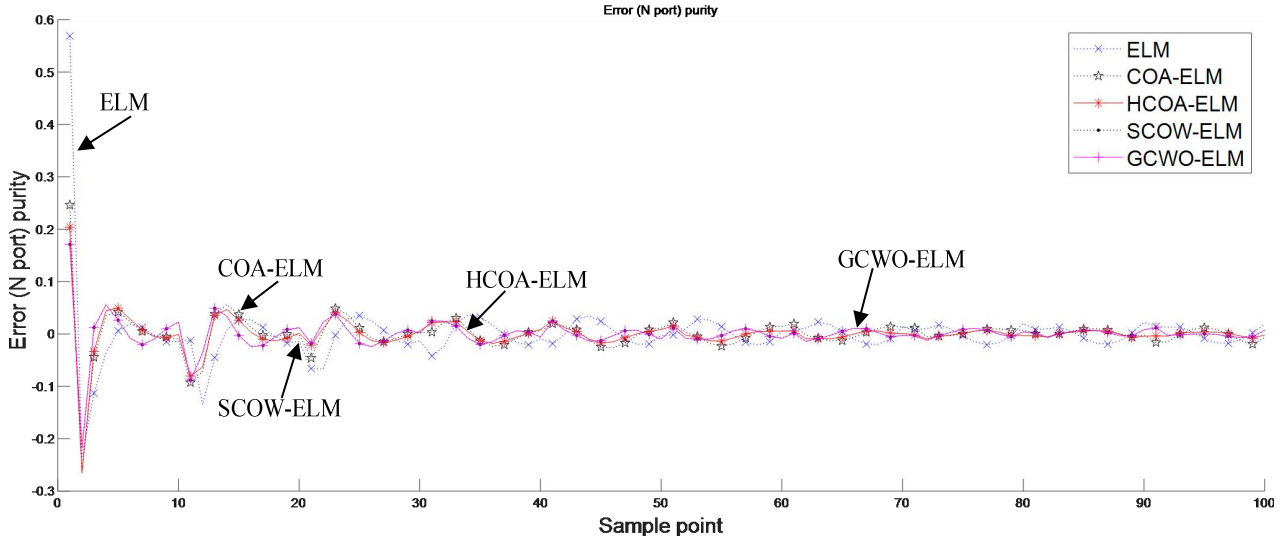

Fig. 8 Prediction error of N-port purity.

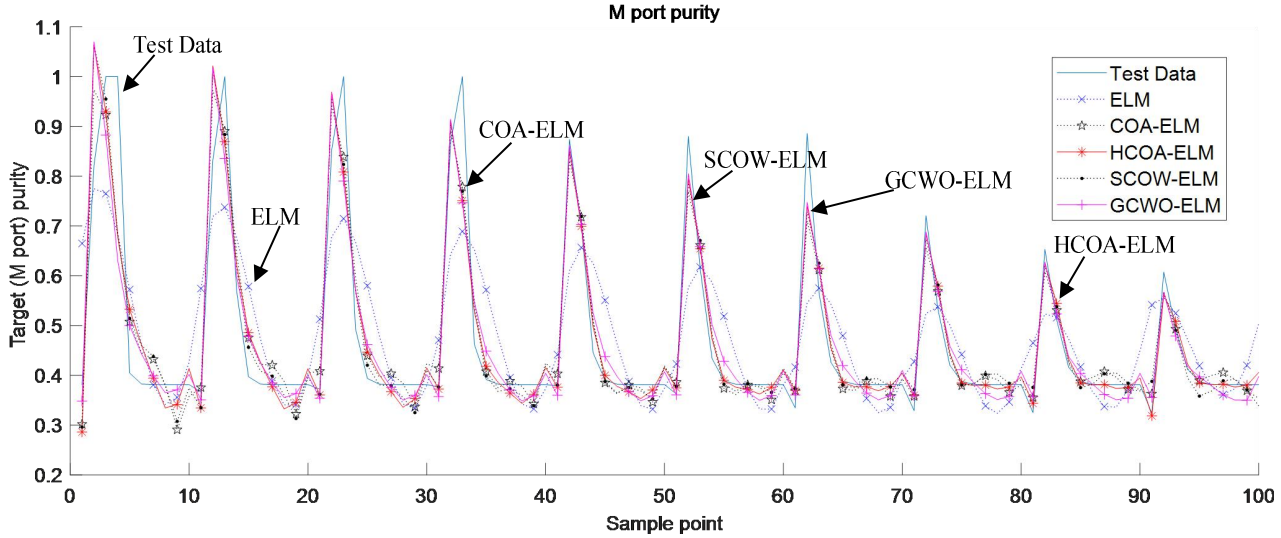

Fig. 9 Prediction results of M-port purity.

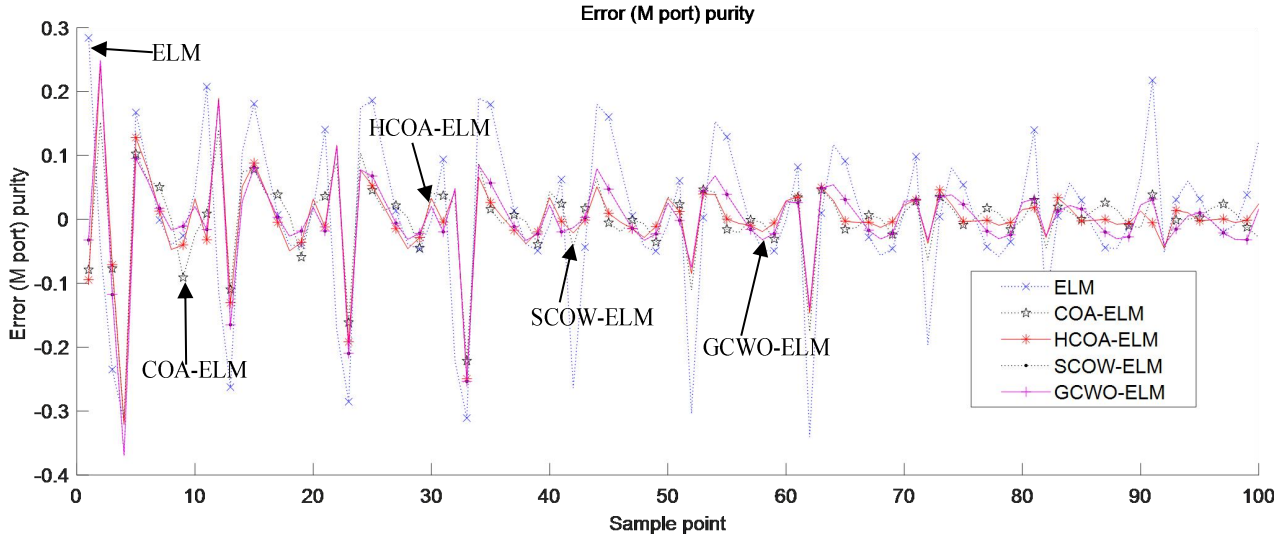

Fig. 10 Prediction error of M-port purity.

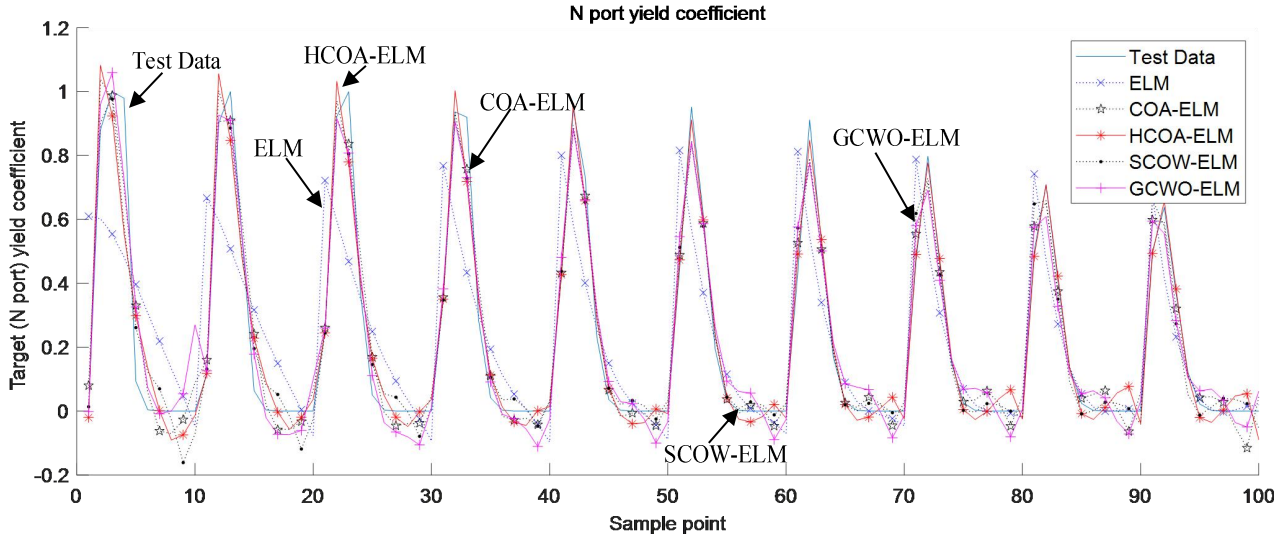

Fig. 11 Predicted yield results of N-port.

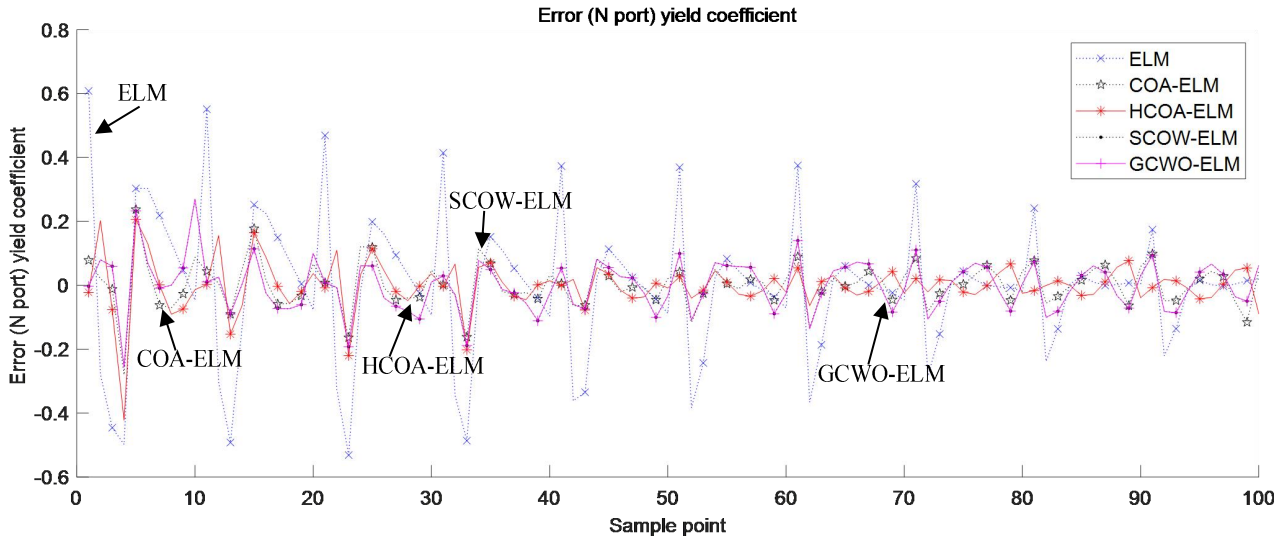

Fig. 12 Yield prediction error of N port.

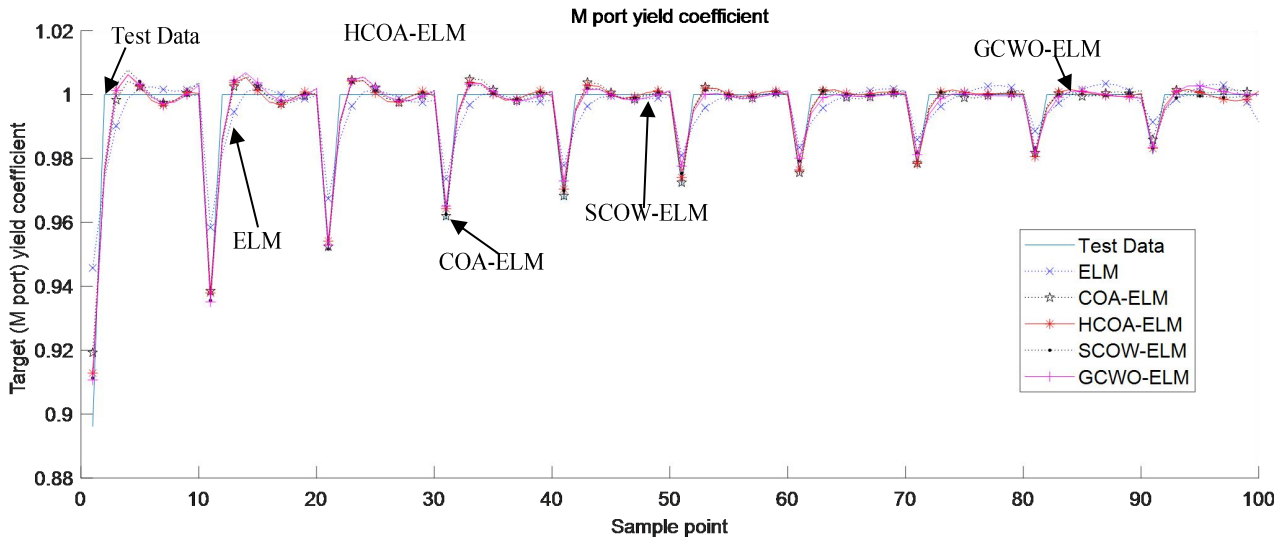

Fig. 13 Prediction results of M-port yield.

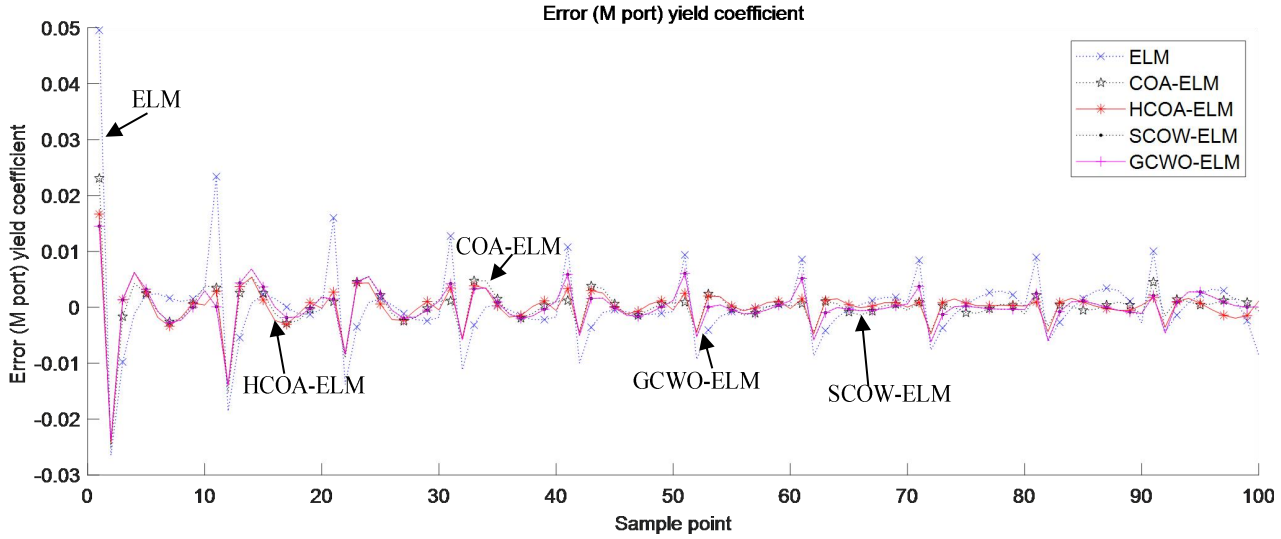

Fig. 14 Prediction error of M-port yield.

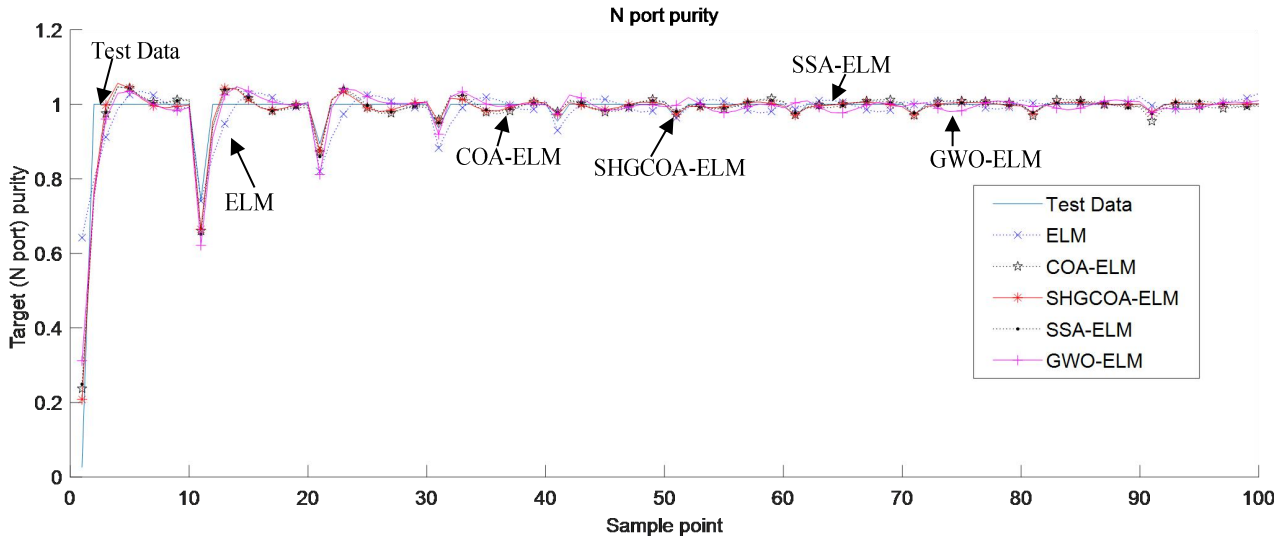

Fig. 15 Purity prediction results of N-port.

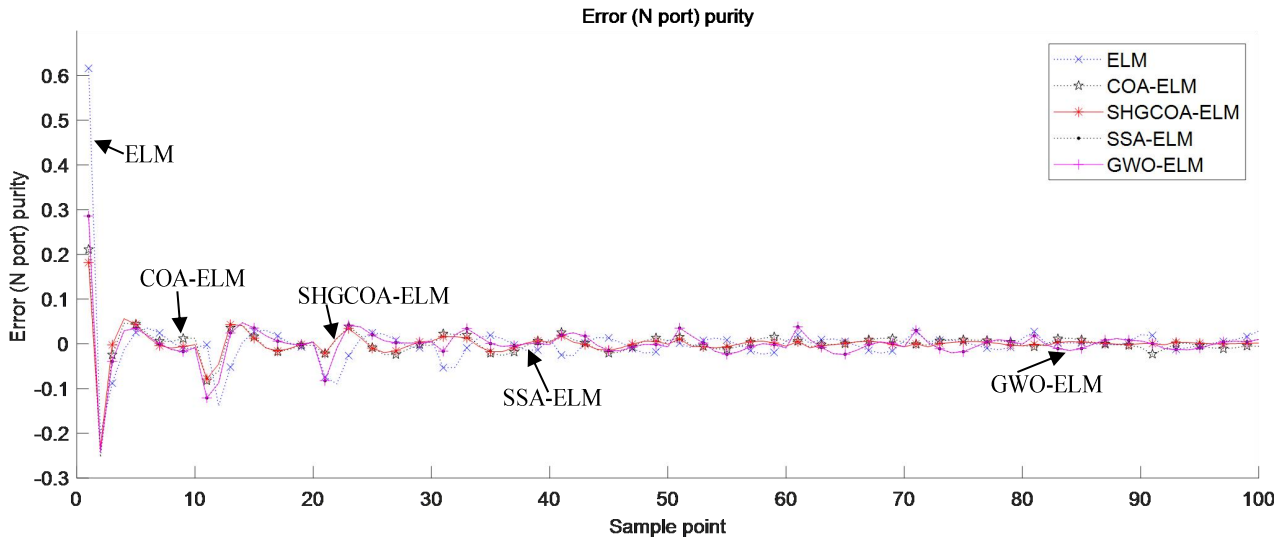

Fig. 16 Prediction error of N-port purity.

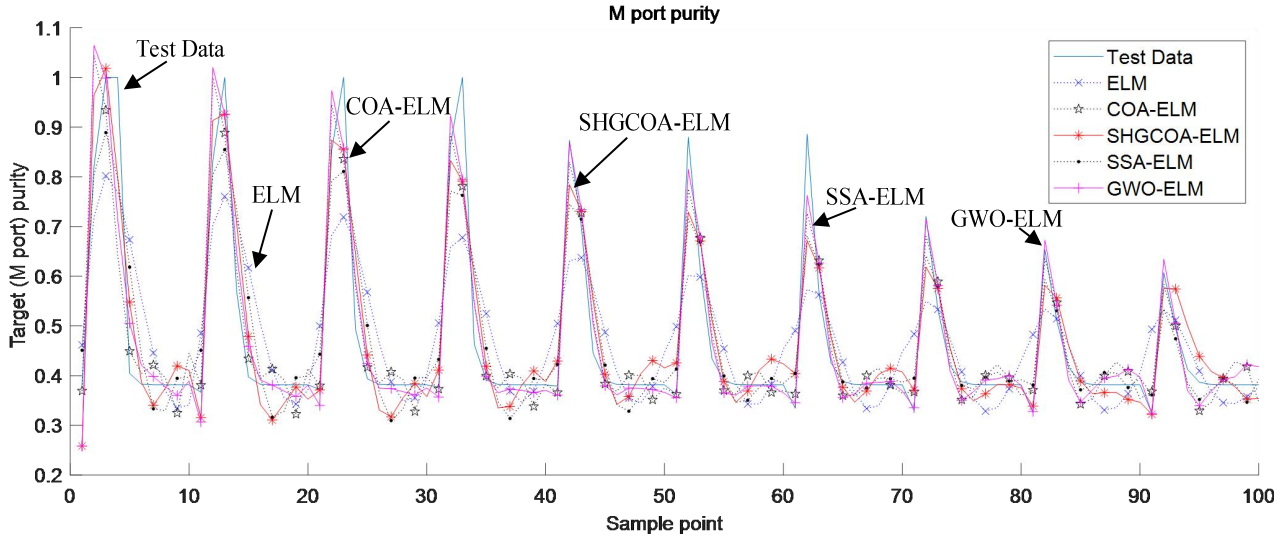

Fig. 17 Prediction results of M-port purity.

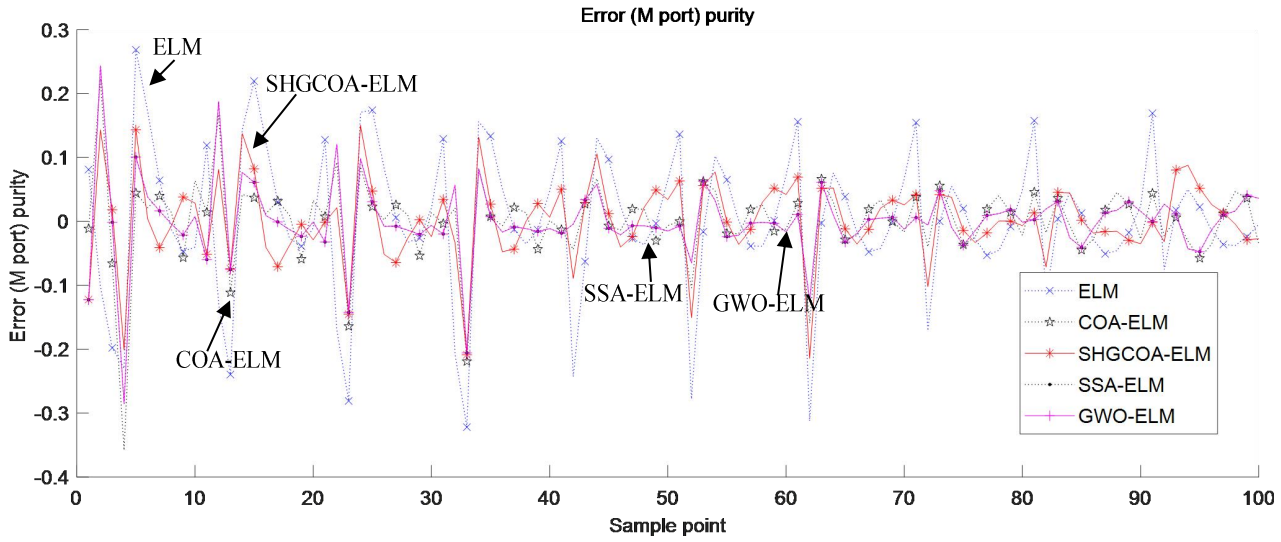

Fig. 18 Prediction error of M-port purity.

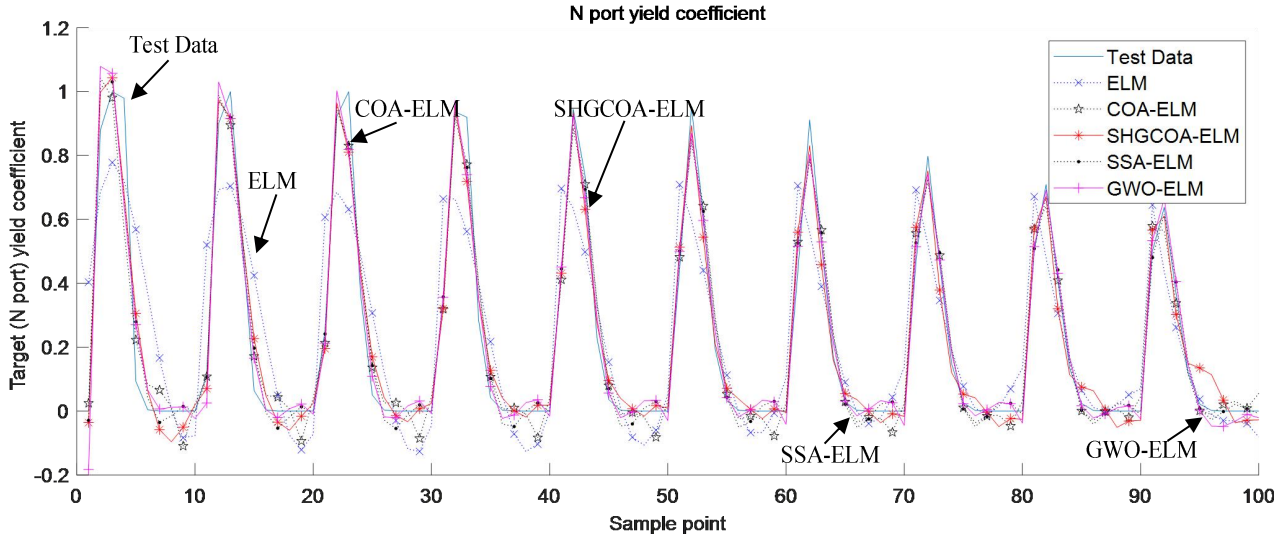

Fig. 19 Prediction results of N yield.

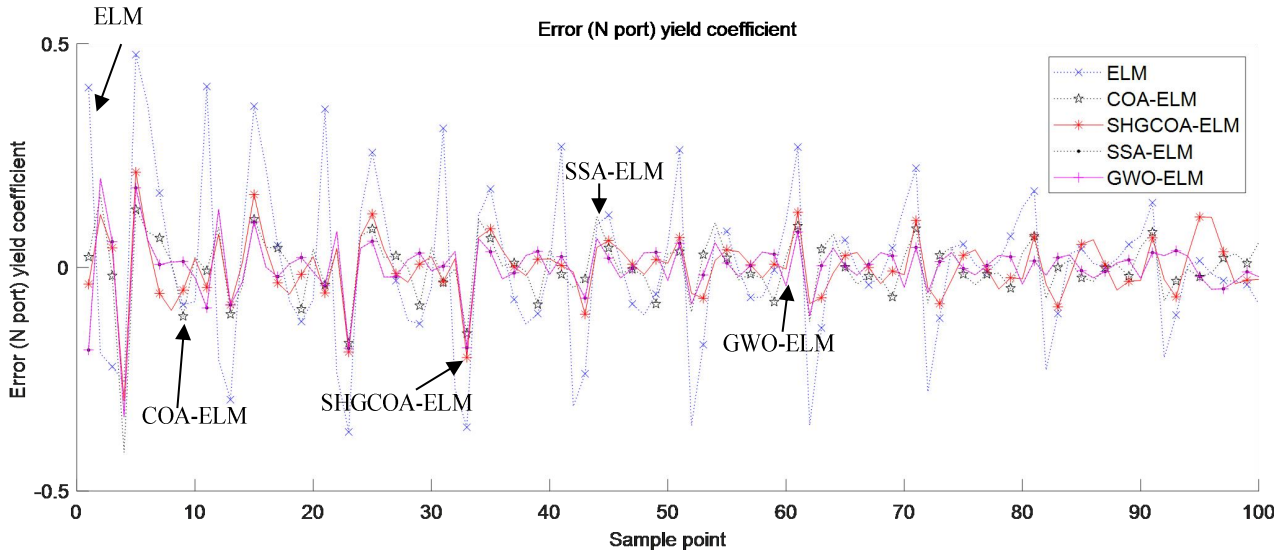

Fig. 20 Prediction error of yield at N port.

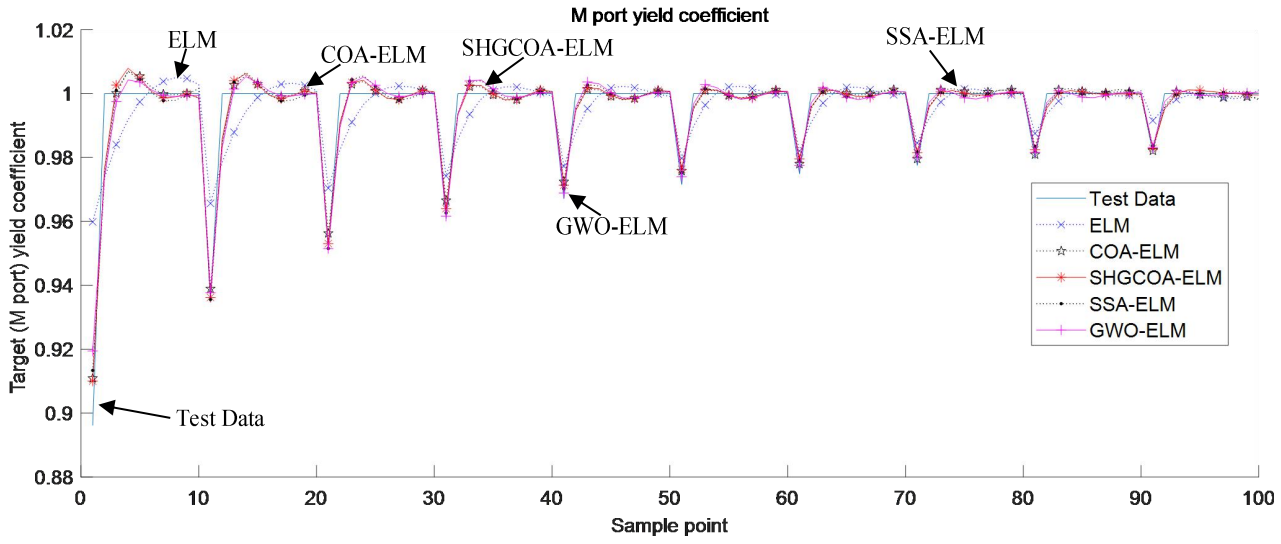

Fig. 21 Prediction results of M-port yield.

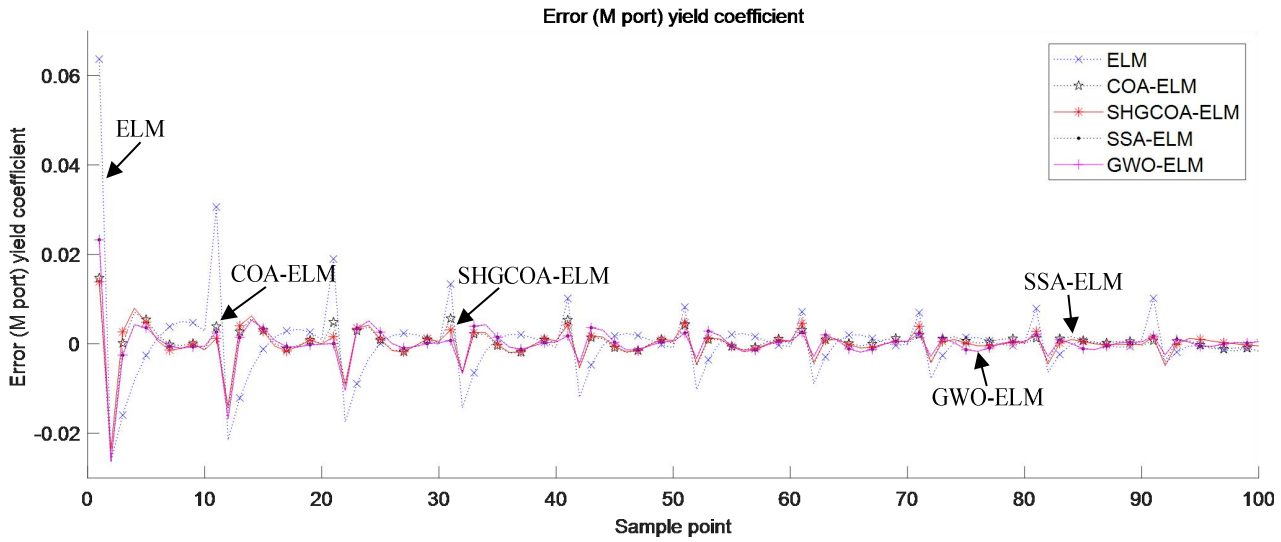

Fig. 22 Prediction error of M yield.

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

Performance index		RMSE	SSE	MAPE	MPE
N-port purity	ELM	0.0670	0.4488	0.0526	0.5619
	COA-ELM	0.0417	0.1741	0.0399	0.2652
	HCOA-ELM	0.0377	0.1420	0.0334	0.2639
	SCOA-ELM	0.0338	0.1139	0.0309	0.2097
	GCOA-ELM	0.0325	0.1055	0.0313	0.2188
M-port purity	ELM	0.1238	1.5318	0.5177	0.3404
	COA-ELM	0.0739	0.5089	0.5508	0.3863
	HCOA-ELM	0.0682	0.4658	0.5435	0.3212
	SCOA-ELM	0.0670	0.4486	0.5413	0.3160
	GCOA-ELM	0.0725	0.5263	0.5405	0.3694
N-port yield	ELM	0.2179	4.7469	0.4150	0.6083
	COA-ELM	0.0799	0.6882	0.3579	0.4798
	HCOA-ELM	0.0755	0.6009	0.3395	0.4221
	SCOA-ELM	0.0794	0.6312	0.5120	0.3995
	GCOA-ELM	0.0805	0.6476	0.2529	0.2703
M-port yield	ELM	0.0079	0.0062	0.0062	0.0496
	COA-ELM	0.0044	0.0020	0.0071	0.0259
	HCOA-ELM	0.0040	0.0016	0.0070	0.0243
	SCOA-ELM	0.0039	0.0015	0.0067	0.0222
	GCOA-ELM	0.0041	0.0017	0.0070	0.0238

TABLE VI. COMPARISON OF PREDICTIVE PERFORMANCE INDEXES OF SOFT-SENSOR MODELS

Performance index		RMSE	SSE	MAPE	MPE
N-port purity	ELM	0.0698	0.4878	0.0474	0.6159
	COA-ELM	0.0442	0.1957	0.0409	0.2857
	SHGCOA-ELM	0.0333	0.1107	0.0285	0.2322
	SSA-ELM	0.0380	0.1443	0.0346	0.2488
	GOW-ELM	0.0375	0.1405	0.0357	0.2531
M-port purity	ELM	0.1150	1.3215	0.5223	0.3219
	COA-ELM	0.0668	0.4461	0.5380	0.3579
	SHGCOA-ELM	0.0674	0.4549	0.5345	0.2145
	SSA-ELM	0.0757	0.5736	0.5291	0.2372
	GOW-ELM	0.0616	0.3798	0.5496	0.2853
N-port yield	ELM	0.1779	3.1665	0.3379	0.4757
	COA-ELM	0.0731	0.5350	0.7268	0.4136
	SHGCOA-ELM	0.0704	0.4951	0.2451	0.2982
	SSA-ELM	0.0637	0.4052	0.3806	0.3161
	GOW-ELM	0.0662	0.4378	0.4032	0.3341
M-port yield	ELM	0.0095	0.0090	0.0069	0.0637
	COA-ELM	0.0049	0.0021	0.0069	0.0265
	SHGCOA-ELM	0.0039	0.0016	0.0067	0.0236
	SSA-ELM	0.0042	0.0018	0.0070	0.0247
	GOW-ELM	0.0045	0.0020	0.0067	0.0262

Fig. 15 shows a comparison of predicted outputs for different purities of the target object in the effluent at port N of the SMB chromatographic separation process. The methods ELM, COA-ELM, SHGCOA-ELM, SSA-ELM and GGO-ELM are used to represent the purities. Fig. 16 illustrates the comparison of prediction errors. In Fig. 17, the predicted outputs for the purity of impurities in the effluent at port M of the SMB chromatographic separation process are shown by using the same methods. The comparison of prediction errors can be seen in Fig. 18. Fig. 19 presents the comparison of predicted outputs for the yield of the target at the N-port of the SMB chromatographic separation process. The methods ELM, COA-ELM, SHGCOA-ELM, SSA-ELM and GOE-ELM are used. The corresponding comparison of prediction errors is displayed in Fig. 20. Moreover, Fig. 21 exhibits the predicted outputs for the yield of impurities at port M in the SMB chromatographic separation process by using the same methods. The comparison of prediction errors can be found in Fig. 22. The simulation experiment chart clearly demonstrates that the soft-sensor model of SMB chromatographic separation, optimized by using the multi-strategy fusion COA, outperforms the single-strategy improved and unoptimized ELM in accurately predicting the purities and yields at both the N-port and M-port. The optimized model exhibits greater effectiveness and accuracy in these predictions.

VI. Conclusion

The soft-sensor model of the SMB chromatographic separation selects input variables including the flow rate of the feed-stock liquid inlet pump (F pump), the flow rate of the flushing liquid inlet pump (D pump) and the valve switching time. The model also considers output variables such as the purity of the target substance in the effluent at port N, the purity of the impurity in the effluent at port M, the yield of the target substance at port N and the yield of the impurity at port M. For optimization of the soft-sensing model in SMB chromatographic separation, both single and multi-strategy fusion algorithms are employed to optimize the ELM model. The simulation results demonstrate that the optimized ELM soft-sensor model with single strategy optimization outperforms the unoptimized model. Furthermore, the multi-strategy fusion COA significantly improves the prediction accuracy of the optimized ELM soft-sensor model, surpassing both the improved single strategy and the unoptimized ELM. Additionally, the established soft-sensor model exhibits strong predictive capabilities for the pecuniary and technical benchmarks in 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. Kim, K. M. Kim, J. W. Lee, S. Kim, F. S. Silva, A. Seidel-Morgenstern, and C. Lee, "Advanced Operating Strategies to Extend the Applications of Simulated Moving Bed Chromatography," *Chemical Engineering & Technology*, vol. 40, no. 12, pp. 2163-2178, 2017.

[4] Y. Zhen, 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. 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, and J. S. Wang, "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, and S. Y. Wang, "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 Accommodating 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] Z. Sen, L. Zheng, H. Xuejiao, and X. Wendong, "A Modified Residual Extreme Learning Machine Algorithm and Its Application," *IEEE Access*, vol. 6, pp. 62215-62223, 2018.

[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, J. Huang, and X. Qiu, "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] X. Wei, 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, H. Su, J. Zhang, and X. Li, "Robust Configuration and Intelligent MPPT Control for Building Integrated Photovoltaic System Based on Extreme Learning Machine," *Journal of Intelligent and Fuzzy Systems*, vol. 33, no. 17, pp. 1-18, 2021.

[17] R. Y. Li, and L. M. Wang, "Soft Measurement Modeling and Chemical Application Based on ISOMAP-ELM Neural Network," *Acta Metrologica Sinica*, vol. 37, no. 5, pp. 548-552, 2016.

[18] S. Zhang, X. Chen, and Y. X. Yin, "An ELM Based Online Soft Sensing Approach for Alumina Concentration Detection," *Mathematical Problems in Engineering*, vol. 2015, pp. 1-8, 2015.

[19] H. F. Pan, A. L. Liu, and D. Automation, "Wavelet Kernel Extreme Learning Machine and Its Application in Soft Sensor Modeling of an Industrial Acetic Acid Distillation System," *Journal of East China University of Science and Technology (Natural Science Edition)*, vol. 40, no. 1, pp. 33-34, 2014.

[20] Y. H. Hang, and Q. B. Hang, "Application of DE-ELM in Soft-Sensing Modeling for Component Concentration of Water Gas," *Process Automation Instrumentation*, vol. 38, no. 6, pp. 63-65, 2017.

[21] C. Zhao, Z. Chen, B. Wang, Y. Wang, and X. Chen, "Soft Sensor Modeling for Reforming Aromatic Hydrocarbon Yield Based on MI and IGSA Optimized ELM," *Chinese Journal of Scientific Instrument*, vol. 40, no. 3, pp. 255-263, 2019.

[22] A. A. Mohammed, R. Minhas, Q. M. J. Wu, and M. A. Sid-Ahmed, "Human Face Recognition Based on Multidimensional PCA and Extreme Learning Machine," *Pattern Recognition*, vol. 44, no. 10-11, pp. 2588-2597, 2011.

[23] Y. H. Kim, H. Hwang, S. Lee, E.-S. Park, S. D. Yoo, J. Lee, J. S. Yang, and M. S. Kwon, "Molecular Cloning and Expression of a Gene for Outer Membrane Protein H in Pasteurella multocida (A:3): Production of Antisera against the OmpH," *Korean Journal of Microbiology & Biotechnology*, vol. 33, no. 4, pp. 274-280, 2005,

[24] R. Minhas, A. Baradarani, S. Seifzadeh, and Q. M. J. Wu, "Human Action Recognition Using Non-separable Oriented 3D Dual-Tree Complex Wavelets," *Asian Conference on Computer Vision*, vol. 5996, pp. 156-160, 2009.

[25] E. Cambria, P. Gastaldo, F. Bisio, and R. Zunino, "An ELM-based model for affective analogical reasoning," *Neurocomputing*, vol. 149, pp. 443-455, 2015.

[26] X. Y. Wang, and M. Han, "Online Sequential Extreme Learning Machine with Kernels for Nonstationary Time Series Prediction," *Neurocomputing*, vol. 145, no. 5, pp. 90-97, 2014.

[27] F. Han, H. F. Yao, and Q. H. Ling, "An Improved Evolutionary Extreme Learning Machine Based on Particle Swarm Optimization," Neurocomputing, vol. 116, pp. 87-93, 2013.

[28] M. Wang, H. Chen, H. Li, Z. Cai, X. Zhao, C. Tong, J. Li, and X. Xu, "Grey wolf optimization evolving kernel extreme learning machine: Application to bankruptcy prediction," *Engineering Applications of Artificial Intelligence*, vol. 63, pp. 54-68, 2017.

[29] B. Wang, K. Y. Wang, D. You, W.H. Chen, and G. Wang, "Real-Time Transient Stability Assessment Based on Genetic Algorithm-Extreme Learning Machine," *Applied Mechanics and Materials*, vol. 427-429, pp. 1390-1393, 2013.

[30] F. Ertam, and E. Avci, "A New Approach for Internet Traffic Classification: GA-WK-ELM," *Measurement*, vol. 95, pp. 135-142, 2017.

[31] S. Zhang, Z. Liu, X. Huang, and W. Xiao, "A Modified Residual Extreme Learning Machine Algorithm and Its Application," *IEEE Access*, vol. 6, pp. 62215-62223, 2018.

[32] L. Chen, J. Zhang, and H. L. Zhang, "Short-Term Photovoltaic Power Generation Forecast Based on VMD-IAA-IHEKLM Model," *Acta Energiae Solaris Sinica*, vol. 44, no. 10, pp. 135-141, 2023.

[33] K. M. Helgen, R. W. Kays, L. E. Helgen, M. T. Tsuchiya-Jerep, C. M. Pinto, K. Koepfli, E. Eizirik, and J. E. Maldonado, "Taxonomic Boundaries and Geographic Distributions Revealed by an Integrative

Systematic Overview of the Mountain Coatis, Nasuella (Carnivora: Procyonidae)," *Biology*, vol. 6, pp. 67-68, 2009.
[34] B. Li, X. Li, H. Rui, and Y. Liang, "Displacement Prediction of Tunnel Entrance Slope Based on Variational Model Decomposition and Grey Wolf Optimized Extreme Learning Machine," *Journal of Jilin University (Engineering and Technology Edition)*, vol. 53, no. 6, pp. 1853-1860, 2023.
[35] X. L. Li, "Research and Application of Improved Salp Swarm Algorithm," M.S. Thesis, University of Science and Technology Liaoning, Anshan, 2023.
[36] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf Optimizer," *Advances in Engineering Software*, vol. 69, no. 3, pp. 46-61, 2014.