Soft-sensing in complex chemical process based on a sample clustering extreme learning machine model^{*}

Di Peng, Yuan Xu, Yanqing Wang, Zhiqiang Geng and Qunxiong Zhu*,**

 * College of Information Science and Technology, Beijing University of Chemical Technology, Beijing 100029, China.
 ** Engineering Research Center of Intelligent PSE, Ministry of Education in China, Beijing 100029, China (e-mail: zhuqx@mail.buct.edu.cn).

Abstract: In actual chemical processes, the fact that some essential variables cannot be directly measured makes the production quality out-of-control and even results in large economic losses. In this study, a novel sample clustering extreme learning machine (SC-ELM) model is developed to achieve timely and accurate measurement. SC-ELM is a fast training algorithm with an excellent generalization performance, and the combined sample clustering approach solves the non-optimal input weights of ELM. The network structure is designed by a fast leave-one-out cross-validation (FLOO-CV) method. Meanwhile, the validity of SC-ELM model is firstly tested by two classical regression datasets. With the comparison of other ELM models, SC-ELM is proved to be an effective model in both modeling accuracy and network structure. Then, SC-ELM is applied in measuring the quality index of a high-density polyethylene (HDPE) process running in a chemical plant, and the experiment results demonstrate that SC-ELM model can achieve quality estimation with higher measuring accuracy and less training time.

Keywords: Extreme learning machine, Density based *K*-means clustering algorithm, Fast leave-one-out cross-validation method, Soft-sensing, High-density polyethylene process.

1. INTRODUCTION

In actual chemical processes, some essential variables that relate to the optimization and control performance are unable or difficult to be directly detected by sensors. To address these issues, the researches of soft-sensing techniques for on-line estimation are extended in past few vears. Currently, some researchers focus on the soft-sensing methods based on the process mechanism analysis or state estimation. These methods require precise mathematical equations of the system, which are often unavailable in practice. In contrast, due to its application of the easily obtained operational data, artificial neural network (ANN) based soft-sensing method is much more suitable for the complex chemical process. However, during the practical application, ANN still exposes a series of questions including the long training time, multiple local minima, and need for tuning of parameters, etc.

Recently, Huang et al. (2006, 2012) invented a fast training algorithm for a single hidden layer feedforward neural network (SLFN), referred as an extreme learning machine (ELM). In comparison with other traditional neural networks, ELM has following advantages: (1) Easy to use and no parameters need to be tuned except predefined network structure; (2) Most ELM network training is accomplished in seconds (so does the large-scale application), which is hundreds of times faster than other traditional training algorithms; (3) ELM has no local minima issues, and it possesses similar approximation performance as BP and SVM; (4) A wide range of activation function, even including some piecewise continuous functions, can be applied in ELM. Even though ELM overcomes the limitations of traditional neural networks, it still has some issues worthy of further study, especially in the chemical application.

One of the essential studies is about the random choosing of input weights and biases. These parameters do not contain any prior knowledge of the inputs and easily cause the hidden layer output matrix not full column rank. This singular hidden layer output matrix sometimes makes the linear system that is used to train output weights unsolvable and also lowers the predicting accuracy. To solve this problem, Miche et al. (2010) proposed an optimally pruned ELM algorithm to remove the redundancy of hidden neurons, Xu and Shu (2006) utilized the global searching ability of particle swarm optimization algorithm to adaptively optimize the input weights, and Feng et al. (2012) updated the ELM by reassigning the hidden neurons with larger weight. However, all of these improvements are just from mathematical calculation, and they ignore the characteristics of training samples. In general, the original data includes some domain knowledge of the modeled object, and the clustering is one of effective ways for extracting domain knowledge. The clustering mechanism establishes groups within the data, assuring that these groups are homogeneous with regard to the output variable. The

^{*} This project is supported by the National Natural Science Foundation of China (No.61104131, No.61473026) and the Fundamental Research Funds for the Central Universities (No.YS1404, No.JD1413).

homogeneity eventually orient the training mechanism into right direction and accelerate the learning itself.

In addition, the preliminary ELM does not provide an effective solution for network structure. In most cases, the optimal number of hidden neurons is selected by a trial and error with the target of minimum training error. This ELM network may perform very well on a similar testing dataset, but when the testing data with subtle differences are introduced, the errors start to dramatically increase. To solve this problem, some cross-validation methods, such as hold-out, K-fold cross-validation (K-CV) and leave-oneout cross-validation (LOO-CV), have been used to find out the optimal ELM structure (see Suresh et al. (2009)). Comparing with the hold-out and K-CV methods, the LOO-CV method can exclude the effects of stochastic variables, and its generalization error estimation is nearly unbiased. Nevertheless, these cross-validation methods have a relatively high computational cost, restricting their applications in some large-scale datasets.

The purpose of this article is to introduce a sample clustering extreme learning machine (SC-ELM) model for estimating the immeasurable variables of chemical processes. The sample clustering is based on a density based *K*means clustering algorithm, and the divided clustering centers are taken as the input weights of ELM. The network structure is rapidly calculated by a fast leave-oneout cross-validation (FLOO-CV) method with the goal of optimal generalization ability. The effectiveness of SC-ELM is firstly verified by 'SinC' and abalone datasets, and then it is used in the quality index estimation of a highdensity polyethylene (HDPE) process. The results show that the SC-ELM efficiently enhances the modeling accuracy of HDPE process, thus exploits a new and effective way to simulate and guide the industry production.

2. REVIEW OF EXTREME LEARNING MACHINE

For N distinct samples $\Omega = \{(\mathbf{x}_i, \mathbf{y}_i) | i = 1, \dots, N; \mathbf{x}_i = [x_{i1}, \dots, x_{in}]^{\mathrm{T}} \in \mathbb{R}^n; \mathbf{y}_i = [y_{i1}, \dots, y_{im}]^{\mathrm{T}} \in \mathbb{R}^m\}$, standard SLFN with L hidden neurons and RBF activation function $\phi_j(\mathbf{x}_i)$ is mathematically modeled as

$$\sum_{j=1}^{L} \boldsymbol{\beta}_{j} \phi_{j} \left(\mathbf{x}_{i} \right) = \sum_{j=1}^{L} \boldsymbol{\beta}_{j} \phi \left(\boldsymbol{\mu}_{j}, \sigma_{j}, \mathbf{x}_{i} \right) = \mathbf{y}_{i}, \quad i = 1, \cdots, N \quad (1)$$

where $\boldsymbol{\beta}_j = [\beta_{j1}, \beta_{j2}, \cdots, \beta_{jm}]^{\mathrm{T}}$ is the weight vector connecting the *j*th hidden neuron and every output neuron, and $\phi_j(\mathbf{x}_i)$ is the output of the *j*th hidden neuron, using the activation function as $\phi_j(\mathbf{x}_i) = \exp\left(-\|\mathbf{x}_i - \boldsymbol{\mu}_j\|^2 / 2\sigma_j^2\right)$. $\boldsymbol{\mu}_j = [\mu_{j1}, \mu_{j2}, \cdots, \mu_{jn}]^{\mathrm{T}}$ is the center of the *j*th hidden neuron, σ_j is its impact width. The above N equations can be written compactly as:

$$\mathbf{H}\boldsymbol{\beta} = \mathbf{Y} \tag{2}$$

where

$$\mathbf{H} = \begin{bmatrix} \phi(\boldsymbol{\mu}_{1}, \sigma_{1}, \mathbf{x}_{1}) & \dots & \phi(\boldsymbol{\mu}_{L}, \sigma_{L}, \mathbf{x}_{1}) \\ \vdots & \cdots & \vdots \\ \phi(\boldsymbol{\mu}_{1}, \sigma_{1}, \mathbf{x}_{N}) & \cdots & \phi(\boldsymbol{\mu}_{L}, \sigma_{L}, \mathbf{x}_{N}) \end{bmatrix}_{N \times L} \begin{bmatrix} \mathbf{H}_{\mathbf{x}_{1}} \\ \vdots \\ \mathbf{H}_{\mathbf{x}_{N}} \end{bmatrix} 3)$$
$$\boldsymbol{\beta} = \begin{bmatrix} \boldsymbol{\beta}_{1}^{\mathrm{T}}, \boldsymbol{\beta}_{2}^{\mathrm{T}}, \cdots, \boldsymbol{\beta}_{L}^{\mathrm{T}} \end{bmatrix}_{L \times m}^{\mathrm{T}} \mathbf{Y} = \begin{bmatrix} \mathbf{y}_{1}^{\mathrm{T}}, \mathbf{y}_{2}^{\mathrm{T}}, \cdots, \mathbf{y}_{N}^{\mathrm{T}} \end{bmatrix}_{N \times m}^{\mathrm{T}} (4)$$

Copyright © 2015 IFAC

ELM proposed by Huang randomly chooses the input weights μ_j and σ_j , then the output weights β are analytically estimated as:

$$\widehat{\boldsymbol{\beta}} = \mathbf{H}^+ \mathbf{Y} \tag{5}$$

where \mathbf{H}^+ is the Moore-Penrose generalized inverse of matrix \mathbf{H} . However, ELM algorithm still has some short-comings in the area of process modeling:

- (1) The random selection of hidden neuron center and impact width is not able to incorporate prior knowledge of the inputs and may contain non-optimum. Thus, the modeling accuracy is decreased.
- (2) The number of hidden neurons is determined by the training error, and this method performs poor when new datasets with subtle changes are introduced.

3. SC-ELM MODEL

In this section, some measures are taken to overcome the above shortcomings of ELM. Firstly, a density based K-means clustering algorithm is set up to partition the training samples into clusters and select the clustering centers as the input weights of ELM. Then, the training set is divided into N groups, and a fast network structure selection method (FLOO-CV) is used to select the optimal number of hidden neurons. Based on the two methods, we conclude the integrated steps of SC-ELM model.

3.1 Density based K-means clustering algorithm

In general, a neural network with higher accuracy requires clusters distributing uniformly while revealing the structure of training samples. Nevertheless, the traditional Kmeans clustering algorithm is easy to achieve a local optimal solution. A consequence of this local optimality is that L clustering centers (L is the number of clusters, equals to the number of hidden neurons) are randomly initialized from training samples, which may result in some centers stuck in nearer regions and never move to where they are needed. Meanwhile, if we choose L training samples with the farthest distance as initial centers, sometimes we will get noisy points, and then it will affect the effect of clustering. In a data space, high-density areas are usually divided by low-density areas, and it is generally accepted that objects in low-density areas are the noisy points.

To avoid getting the noisy points, we select L training samples with the farthest distance in high-density set as the initial clustering centers, where the high-density set is defined as:

Definition 1: ε -neighborhood. The ε -neighborhood is an area of round, which takes the training sample as the center and the distance ε as the radius.

Definition 2: Density of each training sample.

$$\rho\left(i\right) = \frac{Num_i}{\pi\varepsilon^2} \tag{6}$$

where $\rho(i)$ represents the density of the *i*th training sample in set Ω , and Num_i is the number of training samples in ε -neighborhood.

Definition 3: High-density set. The threshold ρ_{\min} is given. If $\rho(i) \geq \rho_{\min}$, the *i*th training sample is added to the high-density set.

With the selected initial clustering centers, the traditional K-means clustering algorithm is adopted to partition the training samples, and obtain the center location of RBF hidden neurons μ_j $(j = 1, 2, \dots, L)$. What is more, the impact width σ_j are described by the nearest distance between center μ_j and other centers, calculating as follows:

$$\sigma_j = \lambda \min \|\boldsymbol{\mu}_j - \boldsymbol{\mu}_w\|, \quad w = 1, 2, \cdots, L, \ w \neq j \quad (7)$$

where λ is the tuning factor, and ranges from 1 to 2.

3.2 Fast leave-one-out cross-validation method

Due to the computational cost of traditional cross validation methods is relatively high, a FLOO-CV method is presented to find the optimal network structure. In FLOO-CV method, the training samples are divided into N groups, shown in Fig.1. Each trial takes out one sample as the validation sample, and its output is determined by the training of the remaining samples. Thus the validation sample, not used for training, stands for the generalization ability of the network.



Fig. 1. Demonstration of leave-one-out cross-validation.

Suppose the sample $(\mathbf{x}_i, \mathbf{y}_i)$ is left out for validation, and the model is trained by remaining samples in set Ω , then the hidden layer output and the desire output of remaining

samples are $\mathbf{H}_{i} = \begin{bmatrix} \mathbf{H}_{\mathbf{x}_{1}}^{\mathrm{T}} \cdots \mathbf{H}_{\mathbf{x}_{i-1}}^{\mathrm{T}} \mathbf{H}_{\mathbf{x}_{i+1}}^{\mathrm{T}} \cdots \mathbf{H}_{\mathbf{x}_{N}}^{\mathrm{T}} \end{bmatrix}_{(N-1)\times L}^{\mathrm{T}}$ and $\mathbf{Y}_{i} = \begin{bmatrix} \mathbf{y}_{1}^{\mathrm{T}} \cdots \mathbf{y}_{i-1}^{\mathrm{T}} \mathbf{y}_{i+1}^{\mathrm{T}} \cdots \mathbf{y}_{N}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$, respectively. The connecting weight vector $\hat{\boldsymbol{\beta}}_{i}$ is firstly obtained by the minimal norm least square solution of equation $\mathbf{H}_{i}\boldsymbol{\beta}_{i} = \mathbf{Y}_{i}$, then the validation output of $(\mathbf{x}_{i}, \mathbf{y}_{i})$ is calculated as:

$$\mathbf{v}_i = \mathbf{H}_{\mathbf{x}_i} \hat{\boldsymbol{\beta}}_i \tag{8}$$

Theorem 1: Suppose there is an output vector $\mathbf{Y}_{i}^{\mathbf{v}_{i}} = \begin{bmatrix} \mathbf{y}_{1}^{\mathrm{T}} \cdots \mathbf{y}_{i-1}^{\mathrm{T}} \mathbf{v}_{i}^{\mathrm{T}} \mathbf{y}_{i+1}^{\mathrm{T}} \cdots \mathbf{y}_{N}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$, so the two equations:

$$\mathbf{H}_i \boldsymbol{\beta} = \mathbf{Y}_i \tag{9}$$

$$\mathbf{H}\boldsymbol{\beta} = \mathbf{Y}_i^{\mathbf{v}_i} \tag{10}$$

have the same minimal norm least square solution.

Proof: Comparing (9), (10) has an extra linear equation:
$$\mathbf{H}_{\mathbf{x}_i}\boldsymbol{\beta} = \mathbf{v}_i \tag{11}$$

Suppose Θ_i and $\Theta_i^{\mathbf{v}_i}$ are respectively the solution space of (9) and (10), and their relationship must be represented as

Copyright © 2015 IFAC

 $\Theta_i^{\mathbf{v}_i} \subseteq \Theta_i$. What is more, \mathbf{v}_i is defined from the minimal norm least square solution $\widehat{\beta}_i$ ($\widehat{\beta}_i \subseteq \Theta_i$), which has been shown in (8), so $\widehat{\beta}_i$ must be the solution in space $\Theta_i^{\mathbf{v}_i}$.

Theorem 2: In the *i*th trial, if there exist inequality $1 - (\mathbf{H}_{\mathbf{x}_i}\mathbf{H}^+)_i \neq 0$, the validation output of the left out sample $(\mathbf{x}_i, \mathbf{y}_i)$ is:

$$\mathbf{v}_{i} = \frac{\mathbf{H}_{\mathbf{x}_{i}}\mathbf{H}^{+}\mathbf{Y} - (\mathbf{H}_{\mathbf{x}_{i}}\mathbf{H}^{+})_{i}\mathbf{y}_{i}}{1 - (\mathbf{H}_{\mathbf{x}_{i}}\mathbf{H}^{+})_{i}}$$
(12)

where $(\mathbf{H}_{\mathbf{x}_i}\mathbf{H}^+)_i$ stands for the *i*th element of vector $\mathbf{H}_{\mathbf{x}_i}\mathbf{H}^+$.

Proof: According to the definition of \mathbf{v}_i and Theorem 1, the following equation is obtained:

$$\mathbf{v}_{i} = \mathbf{H}_{\mathbf{x}_{i}}\hat{\boldsymbol{\beta}}_{i} = \mathbf{H}_{\mathbf{x}_{i}}\mathbf{H}^{+} \begin{bmatrix} \mathbf{Y} - \begin{bmatrix} \mathbf{0}_{(i-1)\times 1} \\ \mathbf{y}_{i} - \mathbf{v}_{i} \\ \mathbf{0}_{(N-i)\times 1} \end{bmatrix} \\ = \mathbf{H}_{\mathbf{x}_{i}}\mathbf{H}^{+}\mathbf{Y} - \left(\mathbf{H}_{\mathbf{x}_{i}}\mathbf{H}^{+}\right)_{i}\mathbf{y}_{i} + \left(\mathbf{H}_{\mathbf{x}_{i}}\mathbf{H}^{+}\right)_{i}\mathbf{v}_{i}$$
(13)

Transpose (13), then the validation output \mathbf{v}_i is as (12).

Therefore, compared with typical LOO-CV method, the proposed FLOO-CV method is not required to undertake N times of model training, and each validation output can be directly calculated according to the (12).

3.3 SC-ELM model construction

Combined with the improved *K*-means clustering algorithm and FLOO-CV method, the model of SC-ELM is constructed. The detailed steps of constructing the SC-ELM model is given as follows:

- (1) Divide the N training samples into N groups, and initialize the number of hidden neurons $l = L^{L}$.
- (2) Train the neural network with all training samples.
- (3) Calculate the training performance. The network output $\left\{ \mathbf{o}_{i} | i = 1, \dots, N; \mathbf{o}_{i} = [o_{i1}, \dots, o_{im}]^{\mathrm{T}} \in \mathbb{R}^{m} \right\}$ has been calculated in step 2. What is more, the performance of neural network is usually evaluated in terms of root mean square (RMSE) criterion, so the training performance is represented as:

$$RMSE_{-}T = \sqrt{\frac{\sum_{i=1}^{N} \sum_{q=1}^{m} (y_{iq} - o_{iq})^{2}}{N \times m}}$$
(14)

- (4) Select each validation sample s (s = s + 1 until s > N) in turn, and calculate its validation output $\mathbf{v}_s = [v_{s1}, \dots, v_{sm}]^{\mathrm{T}} \in \mathbb{R}^m$. If the inequality $1 (\mathbf{H}_{\mathbf{x}_s}\mathbf{H}^+)_s \neq 0$ is satisfied, the validation output \mathbf{v}_s is directly calculated by (12); otherwise, train the SC-ELM with remaining samples, and obtain \mathbf{v}_s by testing the trained SC-ELM with sample s.
- (5) Calculate the validation performance. The validation performance is defined by the following equation:

$$RMSE_V = \sqrt{\frac{\sum_{i=1}^{N} \sum_{q=1}^{m} (y_{iq} - v_{iq})^2}{N \times m}}$$
(15)

804

- (6) Judge whether the criterion of early stopping is met. If the error of RMSE_V increases for a specified number of consecutive iterations (here 5 iterations), indicate the minimum RMSE_V is found in advance, and then turn to Step 8. Otherwise, go to next step.
- (7) Judge whether the number of hidden neurons exceeds tuning range or not. If $l < L^{U}$, set l = l + 1, s = 0, and turn to Step 2. Otherwise, go to next step.
- (8) Plot *RMSE_V* versus the number of hidden neurons, and the plot of *RMSE_V* will have a minimum, which corresponds to an optimal number of hidden neurons.
- (9) On the basis of the determined network structure, re-apply the density based K-means clustering algorithm to decide the RBF center μ and impact width σ , and calculate the output weight $\hat{\beta}$ by using eq. (5). Thus, the model of SC-ELM is completed.

4. PERFORMANCE VERIFICATION

In this section, the performance of SC-ELM model is firstly tested by two classical regression problems: the 'SinC' function and the abalone dataset. The abalone dataset is from UCI machine learning repository, which includes 1 integer and 7 continuous input attributes and 1 integer output. The 'SinC' dataset is generated as follows:

$$y(x) = \begin{cases} \sin(x)/x + \varepsilon, \ x \neq 0\\ 1 + \varepsilon, \qquad x = 0 \end{cases}$$
(16)

where the inputs x are randomly distributed on the interval (-10,10), and the noises ε are uniformly distributed in [-0.2,0.2], adding to all training samples while remaining testing samples noise-free. For the 'SinC' problem, both the training set and testing set have 5000 samples. For the abalone problem, the dataset are divided into 2000 training samples and 2177 testing samples.

Comparing with ELM, SC-ELM uses the prior knowledge (the density based K-means clustering algorithm) to determine the input weights, and it should have a higher predicting accuracy. To prove the superiority of SC-ELM, Table 1 lists the training and testing performance of the standard ELM in the literature (see Huang et al. (2006)) and SC-ELM under the same number of hidden neurons. As observed from Table 1, no matter in 'SinC' dataset or in abalone dataset, SC-ELM obtains smaller training and testing RMSE than ELM in both 'SinC' and abalone dataset. Moreover, standard ELM conducts 50 trials, while SC-ELM only need to train once, so SC-ELM spend less training time than ELM.

 Table 1. Comparison of training and testing

 RMSE of ELM and SC-ELM

Dí	'SinC'		Abalone		
Performance	ELM	SC-ELM	ELM	SC-ELM	
Hidden neurons number	20	20	25	25	
Training RMSE	0.1148	0.1166	0.0803	0.0767	
Testing RMSE	0.0097	0.0074	0.0824	0.0774	
Training Time (s)	0.125×50	0.561	0.0125×50	0.0878	

The structure of SC-ELM model is determined by the FLOO-CV method. Fig.2 shows the training, validation and testing values with the variation of hidden neurons number. For 'SinC' dataset, both the minimum validation

RMSE and the minimum testing RMSE occur at 11 hidden neurons. For abalone dataset, the selected number of hidden neurons by minimum validation RMSE is 20, and its corresponding testing RMSE is 0.0776, which is so close to the minimum testing RMSE occurs at 26 hidden neuron with the value of 0.0773. Therefore, both the two datasets demonstrate the effectiveness of FLOO-CV method.



Fig. 2. Variation in the training, validation and testing RMSE against number of hidden neurons.

In addition, Table 2 lists the performance of many other ELM models in the literature (see Huang et al. (2006); Feng et al. (2012); Deng et al. (2009); Lan et al. (2010)). Both BELM and ES-ELM models improve the randomness property of input weights, and regularized ELM model improves the training algorithm of output weight. Comparing with these three models, SC-ELM has the smallest testing RMSE and smallest hidden neurons in 'SinC' dataset. In the abalone dataset, I-ELM, EM-ELM and CS-ELM models are the methods of determining the number of hidden neurons. Aim at eliminating the influence of random input weights, these three models conducted multiple trials, and the listed hidden neuron numbers are the averages of multiple trials. Furthermore, I-ELM randomly adds the hidden neuron one-by-one without any selection method, so its hidden neuron number reaches extremely large when the training error becomes less than the expected one. As observed from Table 2, the testing RMSE of SC-ELM is only worse than CS-ELM, and the structure of SC-ELM is only more complex than EM-ELM model. Therefore, based on the verification of 'SinC' and abalone datasets, we can conclude that the developed SC-ELM model not only effectively raises the modeling accuracy, but also optimizes the network structure.

Table 2. Performance comparison in 'SinC' and abalone dataset

'SinC' dataset			Abalone dataset		
Models	Hidden	Testing	Models	Hidden	Testing
	Neurons	RMSE		Neurons	RMSE
ELM	20	0.0097	ELM	25	0.0824
BELM	16	0.0139	I-ELM	1799.8	0.0822
ES-ELM	16	0.0134	EM-ELM	11.5	0.0794
RELM	20	0.0078	$\operatorname{CS-ELM}$	20.8	0.0771
SC-ELM	11	0.0054	$\operatorname{SC-ELM}$	20	0.0776

5. HDPE PROCESS APPLICATION

5.1 Description of HDPE process

HDPE cascade reaction (Zhu et al. (2012)) is a polymerization process from high-purity ethylene monomer to highdensity polyethylene under the condition of low-pressure and hexane slurry. It consists of two slurry reactors, post reactor and recycle unit, in which the two slurry reactors are connected by flash tank in series. The flowchart of the polymerization process is shown in Fig.3.



Fig. 3. The flowchart of HDPE cascade reaction process.

In the first slurry reactor, temperature and pressure are firstly set up, and then hexane solvent, catalyst and cocatalyst with suitable ratio, and hydrogen and high-purity ethylene mixed by dehydration and impurity removal are injected. By regulating the feed flow, controlling the ratio between hydrogen and ethylene, and using the effect of catalyst, slurry polymerization is made in the hexane solvent system. Moreover, the heat of polymerization is removed by the latent heat of vaporization of hexane solvent, cooler water in the jacket and external reflux of slurry. The product from the reactor 1 is flash-evaporated to recycle some micromolecule hydrocarbon, and then, is collected into the second reactor to make further polymerization. In the second slurry reactor, hexane solvent, catalyst and cocatalyst need to be supplied under definite temperature and pressure, and then 1-butene is added to regulate the density of the polymer. By regulating the feed flow, controlling the ratio between hydrogen and ethylene, and the ratio between 1-butene and ethylene, new slurry polymerization is made in the hexane solvent system. Later, the polymer suspension is sent to post reactor. In the post reactor and feed vessel, products and solvents are separated, and the final HDPE products are extracted.

The product specification of polyethylene is measured by the melt index of reactor 1 MI_1 , the melt index of reactor 2 MI_2 , and the density of reactor 2 ρ . As lack of online measurement methods in the actual production, the off-line laboratory analysis of the three quality indicators exist a long time delay, which directly influence the control performance of production system as well as polyethylene quality examination and on-line optimization. To address the problem, the proposed SC-ELM model is exploited to on-line estimate the values of three quality indicators.

Copyright © 2015 IFAC

Here three SC-ELM models have been constructed, and three quality indicators, MI_1 , MI_2 , and ρ , are the output of each model.

Based on the sensitivity analysis in the literature (see Zhu and Lang (2011)), the input variables of MI_1 are mainly seven variables in reactor 1: feed flow of ethylene x_1 , feed flow of catalyst x_2 , temperature x_3 , pressure x_4 , partial pressure of ethylene x_5 , partial pressure ratio of hydrogen and ethylene x_6 , and feed flow of hydrogen x_7 . Similarly, MI_2 mainly depends on following six variables in reactor 2: feed flow of ethylene x_8 , feed flow of catalyst x_9 , temperature x_{10} , pressure x_{11} , partial pressure of ethylene x_{12} , and partial pressure ratio of hydrogen and ethylene x_{13} . In addition, two other variables, the pressure in flash tank x_{14} and the melt of index 1 (because the polymer in the reactor 1), also influence the magnitude of MI_2 .

In the HDPE production process, the density of reactor 1 do not need to be predicted since the key material 1-butene is only injected in reactor 2. The density of reactor 2 ρ is regulated not only by the process variables x_8 , x_9 , x_{10} , x_{11} , x_{12} , x_{13} and x_{14} used in the model of MI_2 , but also by the following three variables in reactor 2: feed flow of 1-butene x_{15} , recycle flow of 1-butene x_{16} , and percentage of gas phase butene x_{17} .

5.3 Modeling and experiment result

In this section, the verified SC-ELM is used in the real HDPE process modeling of grade 9455F running in a Chinese petrochemical company.

In order to construct the SC-ELM model, historical online measured input data and off-line analyzed output data of grade 9455F with 10 minutes sample intervals are collected. The data-preprocessing technology, including the methods of handling missing value, data filtering and steady-state identification, is firstly applied to eliminate errors, noises, in-consistent data or missing data. Then, the correlation coefficient matrix (Kashani and Shahhosseini (2010)) is carried out to find the lag time between the output variables and input variables. As a result, the relation between the process variables and three outputs at time t can be described as $MI_1(t) = f(X_1(t-1)), MI_2(t) =$ $f(X_2(t-2), MI_1(t-1))$ and $\rho(t) = f(X_2(t-3), X_3(t-3)),$ where $X_1 \in \{1, 2, 3, 4, 5, 6, 7\}, X_2 \in \{8, 9, 10, 11, 12, 13, 14\}$ and $X_3 \in \{15, 16, 17\}$. Through the above two steps, 626 instances for melt index of reactor 1, 521 instances for melt index of reactor 2 and 351 instances for density of reactor 2 are collected, and these instances of each indicator are divided into training set and testing set in a 2:1 proportion. Moreover, the input weights of SC-ELMs are trained by the density based K-means clustering algorithm, and the network structures are determined by FLOO-CV method.

Fig.4 carries out the variation of validation error with the number of hidden neurons. In Fig.4, the minimum validation RMSE of MI_1 , MI_2 and ρ occurs at the 32, 27 and 29 hidden neurons. Therefore, the model structures of melt index of reactor 1, melt index of reactor 2 and density of reactor 2 are 7-32-1, 8-27-1 and 10-29-1, respectively.



Fig. 4. Variation of validation RMSE against number of hidden neurons.



Fig. 5. Regression analysis between desire and predicted values.

Fig.5 shows the regression analysis between the desire and predicted values in the test set. The regression plot measures the strength of correlation between the predicted network output and the desire network output, and the correlation coefficient (R^2) is calculated as follows:

$$R^{2} = \left[\frac{\sum_{N} (o_{i} - o_{ave})(y_{i} - y_{ave})}{\sqrt{\sum_{N} (o_{i} - o_{ave})^{2}} \sqrt{\sum_{N} (y_{i} - y_{ave})^{2}}}\right]^{2}$$
(17)

where N is the number of instances, o_i is the predicted output values, y_i is the desired output values, and o_{ave} and y_{ave} are the average value of o_i and y_i . As it is evident from Fig.5, both the slopes and the correlation coefficients are close to 1, pointing out the three SC-ELM models are feasible to estimate the HDPE production quality.

Table 3 compares the performance of RBF, ELM and SC-ELM. As seen from Table 3, the generalization relative errors of MI_1 , MI_2 and ρ caused by SC-ELM are 1.11%, 2.21% and 0.010% respectively, obviously better than the error caused by RBF or ELM. For the training time, SC-ELM also runs much faster than RBF or ELM in each indicator. Therefore, this result proves that SC-ELM is more suitable to be the model of dynamic and nonlinear HDPE process. After off-line proving, these established SC-ELM models are then used to calculate the three quality indicators on-line by inputting the field measured data of grade 9455F.

Moreover, the established SC-ELM models are only available for the grade 9455F. If the process grade changes, the property and working point of HDPE are also moving. At this point, new SC-ELMs should be modeled for the new process grade.

Copyright © 2015 IFAC

Table	3.	Performance	Comparison	between
		RBF, ELM a	nd SC-ELM	

Model	General	Generalization relative error			Training time (s)		
	MI_1	MI_2	ho	MI_1	MI_2	ρ	
RBF	2.34%	3.29%	0.015%	7.39	5.53	3.86	
ELM	1.17%	2.67%	0.012%	1.09	0.89	0.91	
SC-ELM	1.11%	2.21%	0.010%	0.95	0.63	0.38	

6. CONCLUSION

To solve the immeasurable variables in complex chemical process, a SC-ELM model has been presented, in which the density based K-means clustering algorithm employs the samples spatial distribution to determine the input weights, and the FLOO-CV method optimizes the neural network structure in a fast and efficient way.

Two classical regression problems, 'SinC' function and abalone age predictions, have been firstly used to evaluate the efficiencies of SC-ELM. With the comparison of other ELM models, SC-ELM is proved to have a fast learning speed, self-organized network structure and excellent modeling accuracy. After certification, SC-ELM model has been applied in the quality index soft-sensing of HDPE process. The estimation result is obviously better than ELM and RBF models with higher accuracy and less training time, thus providing a new way to enhance the production efficiency and ensure the polyethylene quality.

REFERENCES

- Deng, W.Y., Zheng, Q.H., and Chen, L. (2009). Regularized extreme learning machine. In *IEEE Symposium on Computational Intelligence and Data Mining (CIDM)*, 389C–395. Nashville, TN.
- Feng, G.R., Qian, Z.X., and Zhang, X.P. (2012). Evolutionary selection extreme learning machine optimization for regression. *Soft Comput.*, 16, 1485–1491.
- Huang, G.B., Zhou, H.M., Ding, X.J., and Zhang, R. (2012). Extreme learning machine for regression and multiclass classification. *IEEE T. Syst. Man. Cy. B.*, 42, 513–529.
- Huang, G.B., Zhu, Q.Y., and Siew, C.K. (2006). Extreme learning machine: theory and application. *Neurocomputing*, 70, 489–501.
- Kashani, M.N. and Shahhosseini, S. (2010). Constructive hidden nodes selection of extreme learning machine for a methodology for modeling batch reactors using generalized dynamic neural networks. *Chem. Eng. J.*, 159, 195–202.
- Lan, Y., Soh, Y.C., and Huang, G.B. (2010). No reference image quality assessment using modified extreme learning two-stage extreme learning machine for regression. *Neurocomputing*, 73, 3028–3038.
- Miche, Y., Sorjamaa, A., Bas, P., Simula, O., Jutten, C., and Lendasse, A. (2010). Op-elm: Optimally pruned extreme learning machine. *IEEE T. Neural Networ.*, 21, 158–162.
- Suresh, S., Babu, R.V., and Kim, H.J. (2009). No reference image quality assessment using modified extreme learning machine. *Appl. Soft Comput.*, 9, 541–552.
- Xu, Y. and Shu, Y. (2006). Evolutionary extreme learning machine - based on particle swarm optimization. Advances in Neural Networks, 3971, 644–652.
- Zhu, Q.X. and Lang, N. (2011). Research on model structure and input variable selection for industry soft sensing. *Control Engineering of China*, 18, 388–392.
- Zhu, Q.X., Zhao, N.W., and Xu, Y. (2012). A new selective neural network ensemble method based on error vectorization and its application in high-density polyethylene (hdpe) cascade reaction process. *Chin. J. Chem. Eng.*, 20, 1142–1147.