ICIC Express Letters
Part B: Applications
Volume **8**, Number **5**, May **2017**

ICIC International ©2017 ISSN 2185-2766
pp. **867–874**

SOFT-SENSOR MODELING OF PVC POLYMERIZING PROCESS BASED ON OUTPUT-LAYER STRUCTURE FEEDBACK ELMAN NEURAL NETWORK

Weizhen Sun[1] and Jiesheng Wang[1,2,*]

[1]School of Electronic and Information Engineering
[2]National Financial Security and System Equipment Engineering Research Center
University of Science and Technology Liaoning
No. 185, Qianshan Road, Anshan 114051, P. R. China
851062813@qq.com; *Corresponding author: wang_jiesheng@126.com

Received October 2016; accepted January 2017

Abstract. *A soft-sensor model based on the output-layer structure feedback (OSF) Elman neural network (NN) is proposed to predict the conversion velocity and conversion rate of vinyl chloride monomer (VCM) in the polyvinylchloride (PVC) polymerization process. In view of the deficiencies of the original Elman NN, such as simple function of the structure layers and not considering the feedback of nodes in the output layer, the feedback is introduced between the output layer and the structure layer to make the whole network more rigorous without complicated structure. Simulation results show that the improved Elman NN can significantly improve the prediction accuracy of the conversion velocity and conversion rate of VCM and meet the real-time control requirements of polymerization reactor production process.*
Keywords: PVC polymerization process, Elman neural network, Soft sensor, Output-layer feedback

1. **Introduction.** Polyvinyl chloride (PVC resin) is one of the plastic varieties which is the first industrialization realization of the plastics in the world and is one of the most widely used polymers [1]. By using the vinyl chloride monomer (VCM) as raw materials, the production of polyvinyl chloride (PVC) resin by suspension polymerization method is a typical intermittent chemical production process. The different VCM conversion has a certain impact on the molecular weight of PVC resin, thermal stability, porosity, the residues of VCM, the absorptivity of plasticizers and processing liquidity [2]. As a result of the immature detection device and the complexity of the suspension polymerization reaction, the vinyl chloride conversion rate and conversion velocity are hard to acquire in real time, so it is difficult to achieve direct closed-loop control. Elman neural network is a typical local recursive neural network with time delay feedback, which introduces the feedback signal and stores the internal states to realize the dynamic mapping function based on the BP neural network. The Elman NN model has been applied in many fields, such as classification of epileptic seizures [3], forecasting wind speed [4], pressure control for emulsion pump station [5], actuator fault diagnosis of autonomous underwater vehicle [6], short-term load forecasting [7] and motor fault detection [8]. The Elman neural network was integrated with the quantum computation to improve the precision based on the approximation and information processing ability of quantum computation [7]. The genetic algorithm (GA) was used to optimize the weights of Elman NN to realize the detection of the motor fault [8]. However, the adoption of the optimization algorithms must increase the complexity of the algorithm and reduce the computational efficiency.

In the standard Elman NN, the function of the structure layers is too simple in that it only likes a delay operator to remember the output value of the hidden layer at the previous moment. On the other hand, Elman NN only takes account of the feedback of

the double hidden layers, without taking account of the feedback of nodes in the output layer. So, in this paper, the feedback is introduced between the output layer and the structure layer in the standard Elman NN to make the whole network more rigorous without complicated structure. Then the comparison simulation experiments are carried out with standard Elman NN to prove the good performance of the proposed OSF Elman NN. The paper is organized as follows. In Section 2, the technique flowchart of the PVC polymerization process is introduced. The output-layer structure feedback (OSF) Elman neural network (NN) is presented in Section 3. In Section 4, the simulation experiments and results analysis are introduced in detail. The conclusion is illustrated in the last part.

2. **Technique Flowchart of PVC Polymerization Process.** A typical PVC polymerization kettle technological process is shown in Figure 1 [2].

In [2], the hot balancing mechanisms of polymerizer and the influenced factors of convention rate of VCM are analyzed in detail. According to characteristics of polymerization process, the factors influencing conversion velocity are initiator concentration, reaction temperature and polymerize degree. So ten process variables related to VCM conversion rate and conversion velocity are selected as secondary variables of soft-sensor modeling. They respectively are kettle temperature (TIC-P101), kettle pressure (PIC-P102), baffle water flow (FIC-P101), jacket water flow (FIC-P102), injection water flow (FIC-P104), seal water flow (FIC-P105), inlet temperature of cooling water (for jacket water and baffles water sharing, TI-P107), outlet temperature of jacket water (TI-P109), outlet temperature of baffle water (TI-P110), inlet temperature of injection water and seal water (namely the outlet temperature of the cold water tank, TIC-WA01).

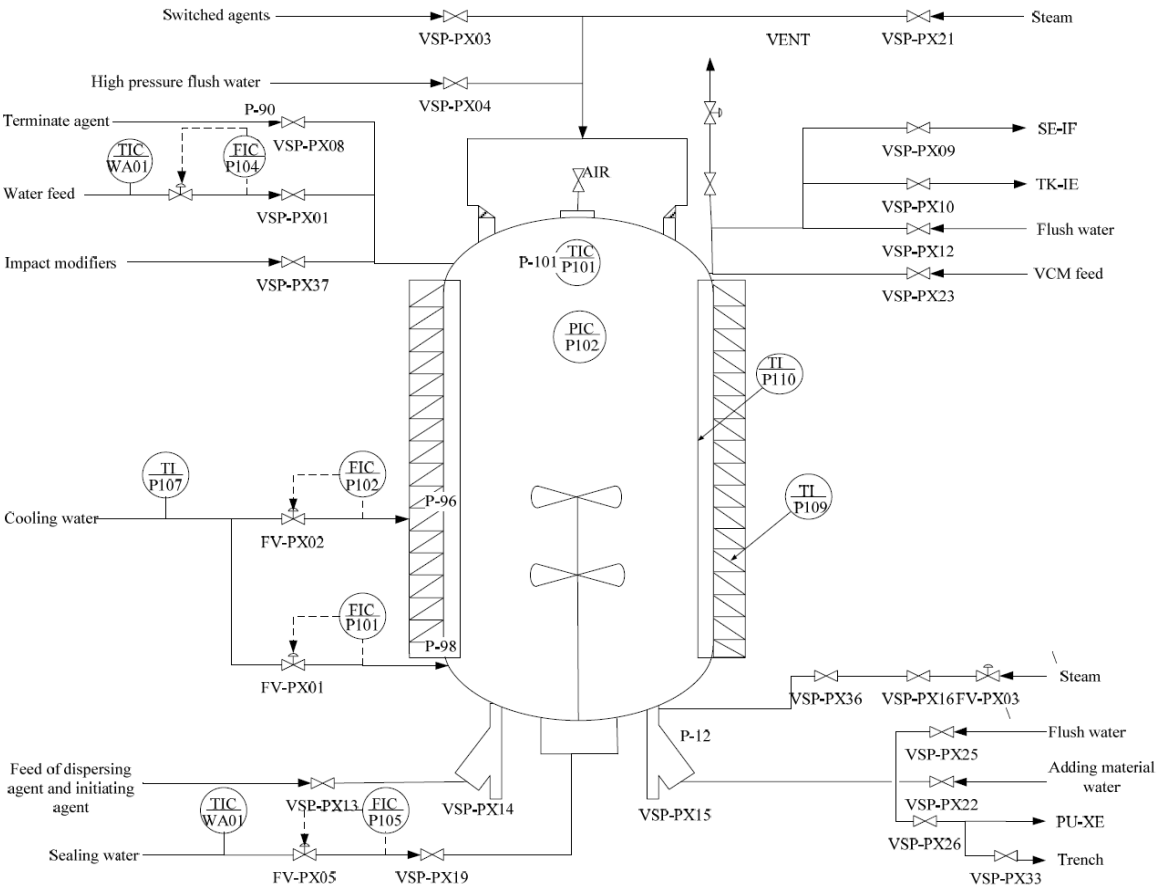
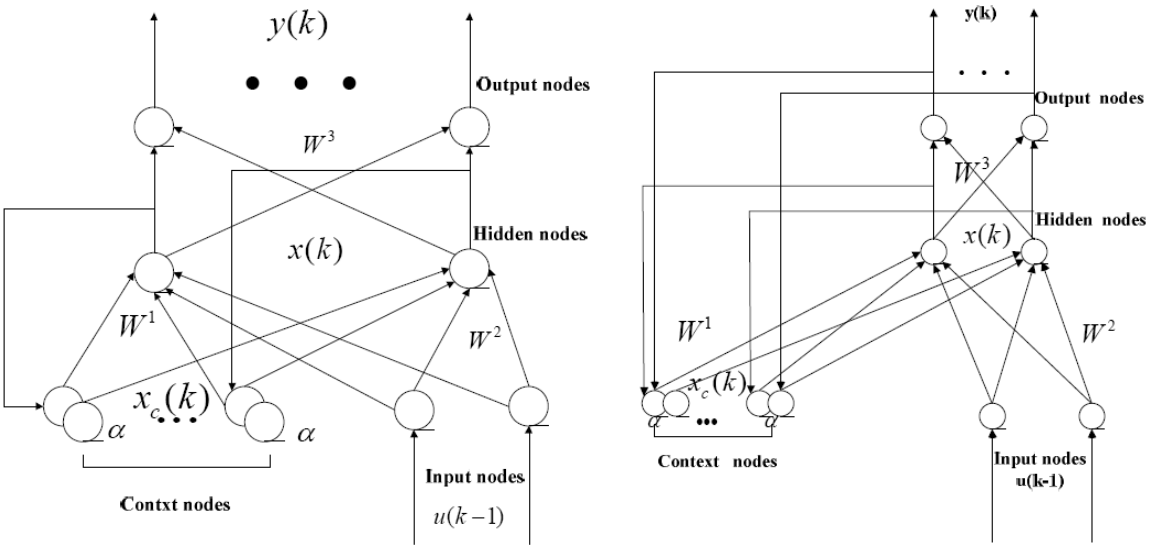


FIGURE 1. Technique flowchart of polymerization kettle

3. Output-Layer Structure Feedback Elman Neural Network.

3.1. OSF-Elman neural network.

Elman NN belongs to a kind of feedback neural network. In order to make the network have the memory function, the feedback is added between the hidden layer and the structure layer. The structure of Elman NN is shown in Figure 2(a). It has a special structural unit besides the input layer, hidden layer and the output layer. The structure of OSF-Elman NN is shown in Figure 2(b).



(a) Elman neural network (b) OSF Elman neural network

FIGURE 2. Structure of Elman neural network and OSF Elman neural network

The addition of the feedback between the output layer and the structural layer in Elman neural network has no effect on the weights between the output layer and the hidden layer and the weights between the input layer and the hidden layer, which does not increase the computation complexity of network. It can be seen from Figure 2(b) that the output in structural layer at k moment is equal to the output in the hidden layer at k-1 moment plus α times of the output in the structural unit at k-1 moment.

$$x_{c,l}(k) = \alpha \cdot x_{c,l}(k-1) + x_l(k-1) \quad l = 1, 2, 3, \ldots, n \tag{1}$$

where $x_{c,l}$ is the output of the lth structure layer, $x_l(k)$ is the output of the lth hidden layer, and α is a self-connected feedback gain factor. So the mathematical model of OSF-Elman NN is shown as follows.

$$x(k) = f\left(W^1(x_c(k) + y(k)) + W^2 u(k-1)\right) \tag{2}$$

$$x_c(k) = \alpha \cdot (x_c(k-1) + y(k-1)) + x(k-1) \tag{3}$$

$$y_k = g\left(W^3 x(k)\right) \tag{4}$$

where W^1 is the weights between structural layer and hidden layer, W^2 is the weights between the input layer and hidden layer, W^3 is the weights between the output layer and hidden layer, $f(x)$ is the Sigmoid activation function of hidden layer neurons, $g(x)$ is a linear activation function of output layer neurons. Their expressions are listed as follows.

$$f(x) = \frac{1}{1 + e^{-x}} \tag{5}$$

$$y_k = W^3 x(k) \tag{6}$$

where y_k is the output of the network model.

3.2. **Learning algorithm of OSF-Elman neural network.** The error function E is defined as:

$$E = \frac{1}{2}(y_d(k) - y(k))^T(y_d(k) - y(k)) \tag{7}$$

where $y_d(k)$ is expected output and $y(k)$ is the real output. The calculation expression $y(k)$ is fed into Equation (7) to obtain:

$$\frac{\partial E}{\partial w_{ij}^3} = -(y_{d,i}(k) - y(k))\frac{\partial y_i(k)}{\partial w_{ij}^3} = -(y_{d,i}(k) - y(k))g_i'(\cdot)x_j(k) \tag{8}$$

where w_{ij}^3 is the weight between the hidden layer and the output layer, $y_{d,i}$ represents the target values and y_i is the predicted output of the model.

Set $\delta_i^0 = (y_{d,i}(k) - y(k))g_i'(\cdot)$ to obtain:

$$\frac{\partial E}{\partial w_{i,j}^3} = -\delta_i^0 x_j(k) \qquad i = 1, 2, \ldots, m \qquad j = 1, 2, \ldots, n \tag{9}$$

where $g_i'(\cdot)$ is the derivative of activation function of the output layer neurons. At the same time, the partial derivative of W^2 is described as follows:

$$\frac{\partial E}{\partial w_{ij}^2} = \frac{\partial E}{\partial x_j(k)}\frac{\partial y_i(k)}{\partial w_{ij}^2} = \sum_{i=1}^{m}\left(-\delta_i^0 w_{ij}^3\right) f_j'(\cdot)u_q(k-1) \tag{10}$$

where w_{ij}^2 is the weight between the input layer and the hidden layer, $u_q(k-1)$ is the input values of the input layer at previous moment and $f_j'(\cdot)$ is the derivative of activation function of the hidden layer.

Alike, set $\delta_j^h = \sum(\delta_j^0 w_{ij}^3) f_j'(\cdot)$ to obtain:

$$\frac{\partial E}{\partial w_{ij}^3} = -\delta_j^h u_q(k-1) \qquad j = 1, 2, \ldots, n \qquad q = 1, 2, \ldots, r \tag{11}$$

At last, the connection weight W^1 from the following layer to the hidden layer is carried out the partial derivation to obtain:

$$\frac{\partial E}{\partial w_{jl}^1} = \sum_{i=1}^{m}\left(\delta_i^0 w_{ij}^3\right)\frac{\partial x_j(k)}{\partial w_{jl}^1} \qquad j = 1, 2, \ldots, n \qquad l = 1, 2, \ldots, n \tag{12}$$

where w_{jl}^1 is the weight between the input layer and the hidden layer. The calculation expression of the $\frac{\partial x_j(k)}{\partial w_{jl}^1}$ is described as follows:

$$\begin{aligned}\frac{\partial x_j(k)}{\partial w_{jl}^1} &= \frac{\partial}{\partial w_{jl}^1}\left[f_j\left(\sum_{i=1}^{h} w_{jl}^1(x_{c,j}(k) + y_i(k-1)) + \sum_{i=1}^{r} w_{jl}^2 u_i(k-1)\right)\right] \\ &= f_j'(\cdot)\left[x_{c,j}(k) + y_i(k-1) + \sum w_{jl}^1\frac{\partial x_{c,j}(k)}{\partial w_{jl}^1}\right]\end{aligned} \tag{13}$$

where $x_{c,j}(k)$ is the output of the jth structural unit. It can be seen from the structure diagram that the basic relationship between $x_c(k)$ and w_{jl}^1 can be ignored. So when the dependence of $x_c(k)$ on the connection weight w_{jl}^1 is not considered, the following equation can be obtained.

$$\frac{\partial x_j(k)}{\partial w_{jl}^1} = f_j'(\cdot)(x_{c,l}(k-1) + y_l(k-1)) \tag{14}$$

where $x_j(k)$ is the output of the jth hidden layer unit

$$f_j'(\cdot)x_{c,j}(k) = f_j'(\cdot)(x_c(k-1) + y_l(k-1)) + \alpha f_j'(\cdot)x_{c,j}(k) \tag{15}$$

Equation (15) is fed into Equation (14) to obtain:

$$\frac{\partial x_j(k)}{\partial w_{jl}^1} = f_j'(\cdot)(x_c(k-1) + y_l(k-1)) + \alpha \frac{\partial x_j(k-1)}{\partial w_{jl}^1} \tag{16}$$

Based on $\Delta W = -\eta \frac{\partial E}{\partial w}$, the learning algorithm of Elman NN can be described as.

$$\Delta w_{ij}^3 = \eta_1 \delta_i^0 x_j(k) \qquad i = 1, 2, \ldots, m; \qquad j = 1, 2, \ldots, n \tag{17}$$

$$\Delta w_{jq}^2 = \eta_2 \delta_j^h u_g(k-1) \qquad q = 1, 2, \ldots, r; \qquad j = 1, 2, \ldots, n \tag{18}$$

$$\Delta w_{ij}^1 = \eta_3 \sum_{i=1}^{m} \left(\delta_j^0 w_{ij}^3\right) \frac{\partial x_j(k)}{\partial w_{jl}^1} \qquad l = 1, 2, \ldots, m; \qquad j = 1, 2, \ldots, n \tag{19}$$

where η_1, η_2 and η_3 are the learning steps of W^1, W^2 and W^3, respectively.

$$\delta_i^0 = (y_{d,i}(k) - y(k)) g_i'(\cdot) \tag{20}$$

$$g_j^h = \sum \left(\delta_i^0 w_{ij}^3\right) f_i'(\cdot) \tag{21}$$

3.3. **Parameter settings of OSF-Elman neural network.** Before adjusting the connection weights, a random initial value to each connection weights is set. Since the system is non-linear, the initial values have a large effect on whether the network converges or falls into local minima or not. It is generally hoped that the initial value of weights can make the state of each neuron close to zero, but not equal to zero or equal to a constant. Otherwise, the system cannot be trained. According to the experimental exploration of a number of scholars, the initial weights are selected as the random number in the scope $(-0.3, +0.3)$, which can be very good to ensure that the network's normal training. During the training process, the learning rate can be automatically adjusted, and the principle can be expressed as follows.

$$\eta(k+1) = \begin{cases} 1.05\eta(k) & SSE(k) > SSE(k-1) \times 1.04 \\ 0.75\eta(k) & SSE(k) < SSE(k-1) \\ \eta(k) & \text{others} \end{cases} \tag{22}$$

where SSE is the sum of the squared errors of the network output. The selection of the initial learning rate $\eta(0)$ is very random, and it is generally initialized as 0.1.

On the other hand, in order to avoid the correction of the connection weight matrix falling into the smallest of energy, the momentum item mc can be added in the correction process of the coefficient matrix of the connection weights. The following equation can be used to adjust the momentum value.

$$mc = \begin{cases} 0 & SSE(k) > SSE(k-1) \times 1.04 \\ 0.95 & SSE(k) < SSE(k-1) \\ mc & \text{others} \end{cases} \tag{23}$$

When $SSE(k) < SSE(k-1)$, it shows that the sum of the network error at this moment is less than the previous one, so the initial value of momentum factor mc is 0.95.

4. **Simulation Experiments.** In this paper, the polymerization industrial process of a chemical factory with 40000 tons/year polyvinyl chloride (PVC) production device is taken as background, whose technology is introduced by American B·F·G company. The input dimension of OSF-Elman network is 10, the output dimension is 1, the number of hidden layer nodes is 25, and the output unit adopts the linear activation function. In order to measure the performances of prediction models, several performance indicators are defined in Table 1, where $\hat{y}$ is predicted value and y is actual value.

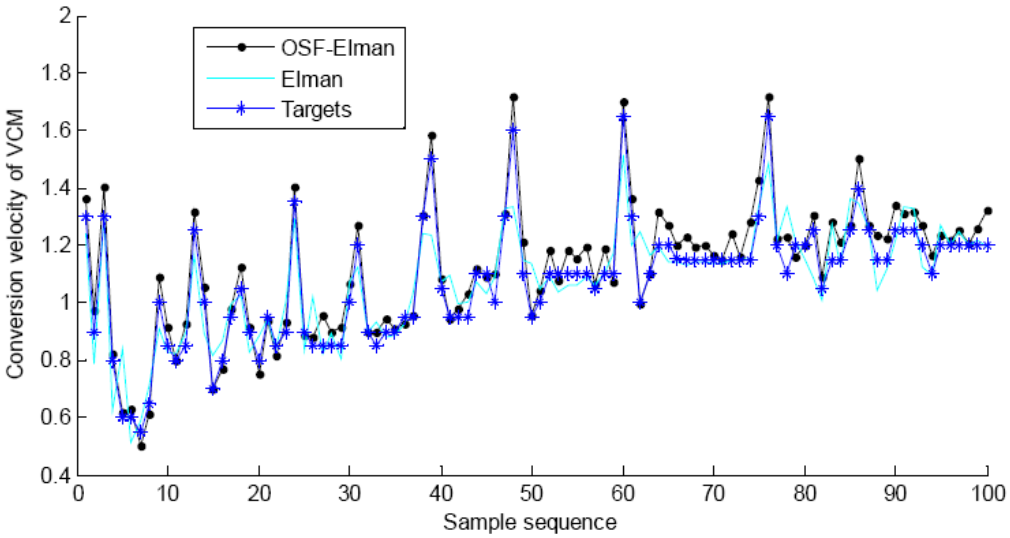
The production historical data of PVC polymerization process are collected and 2 kettle including 1600 sets historical data with the uniformity and representativeness are chosen. Then after data preprocessing the data is divided into two parts, the front 1500 sets

data is the training data, and the rest 100 sets data is used to validate the performance of soft-sensor model. The simulation results are shown in Figure 3 and Figure 4. They show the comparison between the standard Elman NN and the OSF-Elman NN to predict the conversion velocity and conversion rate of the VCM, respectively. The performance comparison results are shown in Table 2.

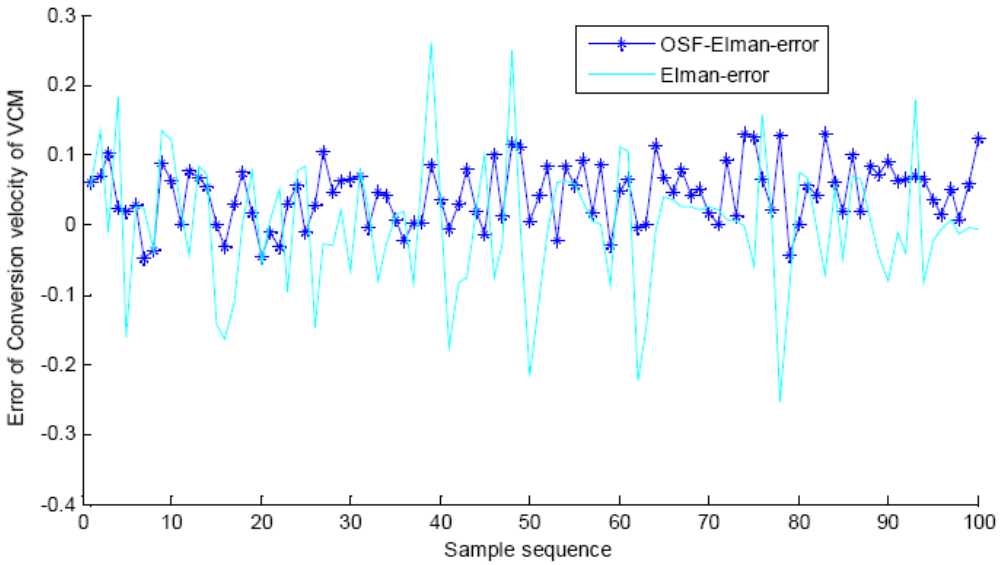
It can be seen from the simulation results that the proposed soft-sensor model based on the OSF-Elman NN is more accurate than the standard Elman neural network. It is

TABLE 1. Definition of model performance index

Maximum positive error	$MPE = \max\{(\hat{y} - y), 0\}$
Maximum negative error	$MNE = \min\{(\hat{y} - y), 0\}$
Root mean square error	$RMSE = \left[\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2\right]^{\frac{1}{2}}$
Sum of squared error	$SSE = \sum_{i=1}^{n}(\hat{y}_i - y_i)^2$

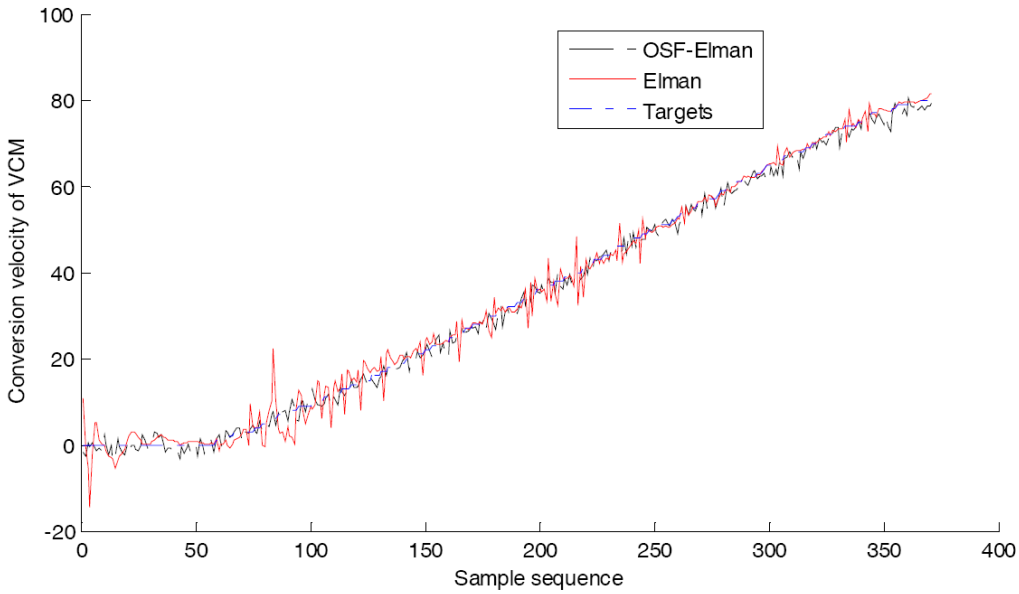


(a) Predictive results

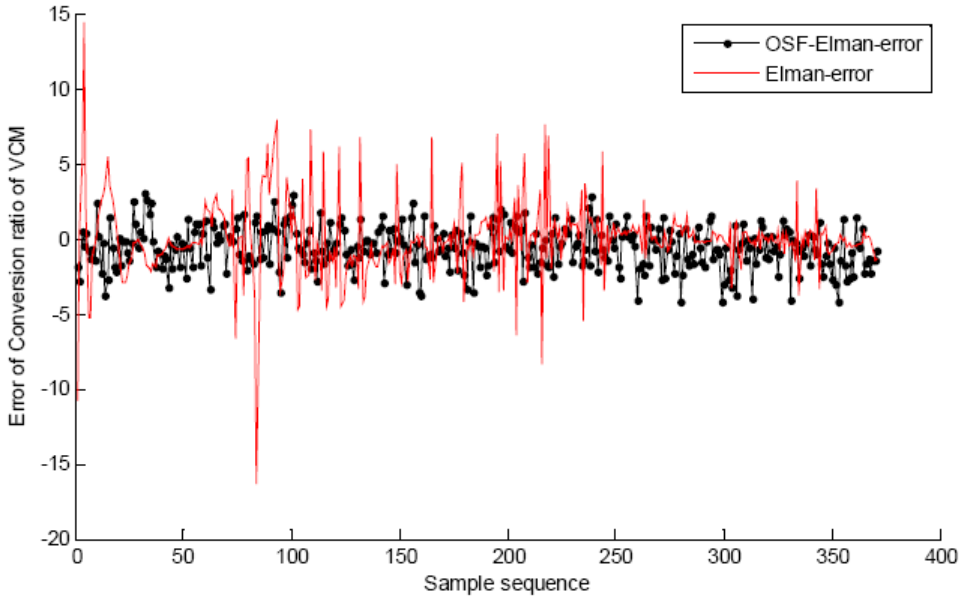


(b) Predictive error

FIGURE 3. Simulation results of conversion velocity of VCM



(a) Predictive results



(b) Predictive error

FIGURE 4. Simulation results of conversion rate of VCM

TABLE 2. Performance comparison results of different soft-sensor models

Predictive variables	Model	MPE	MNE	SSE	RMSE
Conversion velocity	Elman	0.1398	−0.0783	0.0040	0.0638
	OSF-Elman	0.0629	−0.0432	0.0003	0.0173
Conversion rate	Elman	14.3965	−16.2978	0.0040	5.0785
	OSF-Elman	3.0617	−4.2171	0.0003	2.9867

important to improve the production capacity and reduce the production cost of VCM by predicting the VCM conversion velocity and conversion rate.

5. **Conclusions and Future Work.** This paper proposes a soft-sensor model of PVC polymerizing process based on the OSF-Elman neural network. The simulation results show that the improved Elman NN has higher accuracy in predicting the conversion

velocity and conversion rate of VCM than standard Elman NN. In future, the swarm intelligent optimization methods will be used to optimize the parameters of Elman NN.

Acknowledgment. This work is supported by the Program for Liaoning Excellent Talents in University (Grant No. LR2014008), the Project by Liaoning Provincial Natural Science Foundation of China (Grant No. 2014020177), and the Program for Research Special Foundation of University of Science and Technology of Liaoning (Grant No. 2015TD04).

REFERENCES

[1] S. Zhou, G. Ji, Z. Yang and W. Chen, Hybrid intelligent control scheme of a polymerization kettle for ACR production, *Knowledge-Based Systems*, vol.24, no.7, pp.1037-1047, 2011.

[2] J. S. Wang, S. Han and Q. P. Guo, Echo state networks based predictive model of vinyl chloride monomer convention velocity optimized by artificial fish swarm algorithm, *Soft Computing*, vol.18, no.3, pp.457-468, 2014.

[3] S. Raghu, N. Sriraam and G. P. Kumar, Classification of epileptic seizures using wavelet packet log energy and norm entropies with recurrent Elman neural network classifier, *Cognitive Neurodynamics*, pp.1-16, 2016.

[4] J. Wang, W. Zhang, Y. Li et al., Forecasting wind speed using empirical mode decomposition and Elman neural network, *Applied Soft Computing*, vol.23, no.5, pp.452-459, 2014.

[5] C. Tan, N. Qi, X. Zhou et al., A pressure control method for emulsion pump station based on Elman neural network, *Computational Intelligence & Neuroscience*, vol.2015, no.5, pp.1-8, 2015.

[6] Y. S. Sun, Y. M. Li, G. C. Zhang et al., Actuator fault diagnosis of autonomous underwater vehicle based on improved Elman neural network, *Journal of Central South University*, vol.23, no.4, pp.808-816, 2016.

[7] Z. Zhang and W. Gong, Short-term load forecasting model based on quantum Elman neural networks, *Mathematical Problems in Engineering*, vol.2016, no.3, pp.1-8, 2016.

[8] X. Z. Gao and S. J. Ovaska, Genetic algorithm training of Elman neural network in motor fault detection, *Neural Computing & Applications*, vol.11, no.1, pp.37-44, 2002.