

التنبؤ بسرعة عبء النسب في فرن الانفجار إستناداً إلى آلة التعلم المدقع

شين قوان*، ييشين بين**، سين تشانغ** وهيجانغ تشانغ**
* كلية هندسة المعلومات، جامعة لينغنان عادي، قوانغدونغ، تشانجيانغ 524048، الصين
** مدرسة الأتمتة والهندسة الكهربائية، جامعة العلوم والتكنولوجيا بكين، بكين 100083

الخلاصة

عبء الشحن هو أهم عملية من العمليات العليا في فرن الصهر. موقف وسرعة طبقة عبء النسب يمكن أن تعكس حالة فرن الانفجار، ويمكن أن توجه المشغلين للعبء المقبل للشحن. في هذه الورقة، يتم تأسيس النسب النسبية للتنبؤ بنموذج طبقة عبء من قبل خوارزمية آلة التعلم المدقع. يمكن للنموذج أن يجعل التنبؤات خطوة واحدة ومتعددة الخطوات لسرعة عبء النسب باستخدام بيانات الرادار الحقيقي ومعلومات الحالة في فرن الصهر. في جزء المحاكاة، جمعنا بيانات الإنتاج الحقيقية في عملية صنع الحديد، وحصلنا على نتائج المحاكاة الدقيقة عن طريق استخدام المخطط المقترح.

Prediction of burden descent speed in blast furnace based on extreme learning machine

Xin Guan*, Yixin Yin**, Sen Zhang** and Haigang Zhang**

* School of Information Engineering, Lingnan Normal University, Guangdong, Zhanjiang 524048, China

**School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China

Corresponding Author: zhangsen@ustb.edu.cn

ABSTRACT

The burden charging is the most important operation of the upper operations in blast furnace. The position and descent speed of burden layer can reflect the situation of blast furnace and can guide the operators in the next burden charging. In this paper, the descent speed prediction model of burden layer is established by extreme learning machine algorithm. The model can make single-step and multi-step predictions to the burden descent speed using real radar data and status information in blast furnace. In the simulation part, we collected the real production data in iron-making process and obtained the satisfied and accurate simulation results by employing the proposed scheme.

Keywords: Burden charging; blast furnace; descent speed; extreme learning machine.

INTRODUCTION

The iron and steel industry is a pillar industry in China. The iron-making process in blast furnace (BF) is the first step in iron and steel industry (Liu, 2011). The BF smelting process, affected by a variety of external factors, is often operating in harsh environments. In addition, because of lack of inspection data, unclear mechanism, and inaccurate model, BF is a typical ‘black box’ system (Jian, 2013). The operation in iron-making process mainly relies on the experience of the operators, resulting in the fact that the BF is generally under ‘reasonable-deteriorative-reasonable’ repeated state. The burden charging operation is the upper part of BF operations. It is a prerequisite to the reasonable burden surface and descent speed of burden layer for the stable running of BF (Luo, 2011). The burden layer distribution is closely related to the gas flow distribution, the fuel ratio, the cohesive zone structure, and so on. The descent model of burden layer is the guidance for the operator to monitor the status of furnace condition and adjust next charging (Gao et al., 2012).

It is important to make guidance for the production of BF based on the prediction model of burden layer descent speed. The operators infer the descent speed of burden layer based on production experience and BF status indicators (as temperature, pressure, etc.), which makes the BF operation empirical and subjective. The mechanism model of burden layer descent has been a hot research topic in academia and industry. Nishio (2007) and Reza (2013) have established the descent model of burden layer, respectively, and presented that the descent speed of burden in the furnace center is 1.2~1.5 times the speed around the furnace wall. With the application of radar technology in the burden surface detection, data-driven technology has been applied in the descent

model of burden surface. Herik has established the BF burden layer distribution model based on single-point radar measurement system (Henrik & Njanhinnela, 2004). In the real BF production, compared to the monitor of descent speed of burden layer, it is more important to make prediction of the descent speed. Accurate speed prediction can help the operator monitor the status inside BF and present guidance to the lower part operation.

Extreme Learning Machine (ELM) algorithm is proposed subject to the single hidden layer neural networks (SLNNs) (Huang et al., 2006; Simone, 2015). Compared with other neural network algorithms, the learning parameters in ELM algorithm can be generated randomly, while the output weights can be obtained by the least squares estimation method (Huang et al., 2011). Since the weights of the hidden layer nodes do not need to be modified, ELM algorithm has fast training speed and can meet the needs of online implementation. Huang has proved the universal approximation capability of ELM model, which promotes the development of the theory of ELM algorithm (Huang et al., 2012). Nowadays, ELM algorithm has been successfully applied in many real world applications such as regression, classification and clustering, and so on (Feng et al., 2009). Meanwhile, there have been many ELM algorithm variants subject to different data forms. OS-ELM algorithm is the online sequential learning version based on ELM model. OS-ELM algorithm can well deal with cases when data arrive chunk-by-chunk (Liang et al., 2006). Here we employ ELM and OS-ELM algorithm in the decline model of the burden surface in BF.

In this paper, the ELM algorithm is employed in the prediction model for burden descent in BF based on the real radar data. The industrial radar detection system for the burden layer position can calculate the descent speed. The prediction speed information will be submitted to the operators for the guidance to the next charging operation. The descent of burden layer can be reflected by the operating indexes of BF. We select some closely related production indexes as the prediction model inputs. In addition, taking into account the acquirement of online running of the prediction model, OS-ELM algorithm is employed in the multi-step prediction model for the burden descent.

The rest of this paper is organized as follows. Section 2 presents a review of the related works including the ordinary ELM and online ELM algorithms. In Section 3, we establish the descent speed prediction model of burden layer and the simulation results will be shown in Section 4. Then Section 5 is the conclusion of our work.

Extreme Learning Machine

ELM theory

Here we present a review of the state-of-the-art ELM algorithm and then we will discuss kernel ELM scheme.

Given a dataset containing N training samples, (x_i, t_i) , $x_i = [x_{i1}, x_{i2}, \dots, x_{in}]^T \in R^n$ denotes an n -dimensional feature of the i th sample and $t_i = [t_{i1}, t_{i2}, \dots, t_{im}] \in R^m$ denotes the corresponding desired output. The mathematical model of ELM with \tilde{N} hidden nodes and activation function $g(x)$ can be summarized as follows (Huang et al., 2006; Teena & Sharma, 2016):

$$\sum_{i=1}^{\tilde{N}} \beta_i g_i(x_k) = \sum_{i=1}^{\tilde{N}} \beta_i g(w_i, b_i, x_k) = t_k, k = 1, 2, \dots, N \quad (1)$$

where $W = [w_1, w_2, \dots, w_{\tilde{N}}]^T$ and $B = [b_1, b_2, \dots, b_{\tilde{N}}]^T$ are the learning parameters generated according to any continuous probability distribution randomly. Such nonlinear piecewise continuous function can be, but not limited to, (Huang, 2015)

(1) Sigmoid function:

$$G(a_j, b_j, x_i) = \frac{1}{1 + \exp(-(a_j \cdot x_i + b_j))} \tag{2}$$

(2) Hard-limit function:

$$G(a_j, b_j, x_i) = \begin{cases} 1, & \text{if } a_j \cdot x_i - b_j \geq 0 \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

(3) Gaussian function:

$$G(a_j, b_j, x_i) = \exp(-b_j \|x_j - a_i\|^2) \tag{4}$$

(4) Multiquadric function:

$$G(a_j, b_j, x_i) = (\|x_j - a_i\|^2 + b_j^2)^{1/2} \tag{5}$$

Then the above can be written in matrix form as

$$H\beta = T \tag{6}$$

where $H(W, B, X) = \begin{pmatrix} g(w_1 \cdot x_1 + b_1) & \dots & g(w_{\tilde{N}} \cdot x_1 + b_{\tilde{N}}) \\ \vdots & \ddots & \vdots \\ g(w_1 \cdot x_N + b_1) & \dots & g(w_{\tilde{N}} \cdot x_N + b_{\tilde{N}}) \end{pmatrix}_{N \times \tilde{N}}$ called the hidden

layer output matrix. According to the theory of least squares, the output weight β can be estimated as

$$\hat{\beta} = H^+ T \tag{7}$$

where H^+ is the Moore-Penrose generalized inverse of H (Henrik et al., 2004).

The model of ELM algorithm is presented in Figure 1, where one can see that ELM would map the input data into \tilde{N} -dimension space named ELM feature space (Huang, 2011). Huang and his colleagues have proved that ELM algorithm has the universal approximation capability; that is to say, ELM has the ability to approach any continuous target function. Hence, it is reasonable to do feature learning, clustering, regression, classification, and so on in ELM feature space. In addition, there are three types of feature maps: compressed feature representation for $N > \tilde{N}$, equal dimension feature representation for $N = \tilde{N}$, and spare feature representation for $N < \tilde{N}$. In addition, we present the commonly used method to calculate the output weights through SVD method in the bottom right corner of Figure 1. In order to get rid of the puzzle of multicollinear problem, a regularization term $1/c$ is added to the diagonal of $H^T H$ or HH^T .

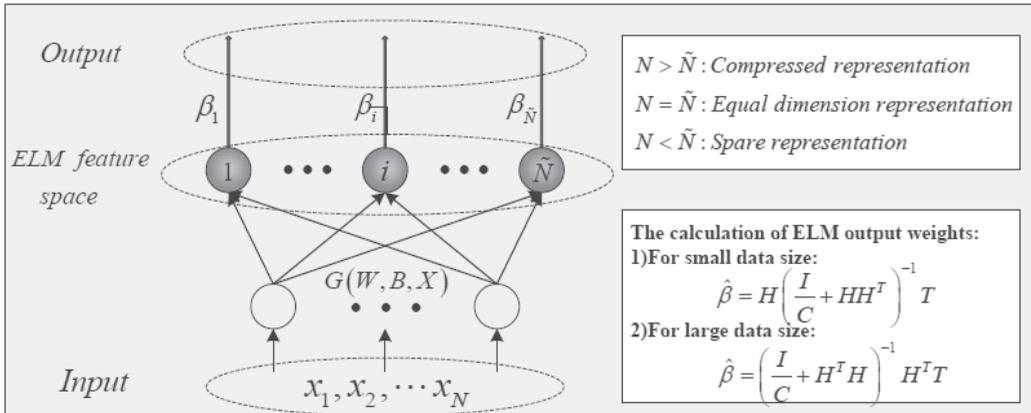


Figure 1. ELM model and ELM feature mapping.

The essence of ELM algorithm is the random weight selection of hidden nodes, which can make sure of the fast training speed. Huang has proved the universal approximation capability as shown in the following.

Universal approximation capability: given any bounded non-constant piecewise continuous function as the activation function in hidden neurons, if by tuning parameters of hidden neuron activation function SLFNs can approximate any target continuous function, then for any continuous target function $f(x)$ and any randomly generated function sequence

$$\{h_i(x)\}_{i=1}^L, \lim_{L \rightarrow \infty} \left\| \sum_{i=1}^L \beta_i h_i(x) - f(x) \right\| = 0, \text{ holds with probability one with appropriate}$$

output weights β .

Online sequential ELM algorithm

As mentioned above, the training data may arrive chunk-by-chunk or one-by-one (a special case of chunk) in real applications. The OS-ELM algorithm aims at online learning cases and constantly updates the output weights within short time (Huang, 2006).

When a new chunk of sampling data comes, the mathematical model of ELM should be modified as

$$\begin{bmatrix} H \\ \delta H \end{bmatrix} \beta' = \begin{bmatrix} T \\ \delta T \end{bmatrix} \tag{8}$$

where δH and δT are the newly generated hidden layer output matrix using the previous learning parameters and output matrix consisting with newly obtained observations, and β' is the modified output weight matrix.

There are two phases in OS-ELM algorithm, an initialization phase and a sequential phase. The initialization phase is the same as the ordinary ELM algorithm, while the output weight matrix β will be updated in the sequential phase through an iterative way. The OS-ELM algorithm can be summarized as follows.

Initialization phase: given a small chunk of training data to initialize the learning, $S_0 = \{(x_i, t_i) | x_i \in R^n, t_i \in R^m, i = 1, 2, \dots, N_0\}$ from the whole training set S . Here it is worth noting that the number of chunks of training data required in the initialization phase N_0 should be equal to or greater than the number of hidden nodes \tilde{N} .

- (1) Randomly assign the learning parameters a_j and $b_j, j = (1, 2, \dots, \tilde{N})$.
- (2) Calculating the initial hidden layer output matrix H_0

$$H_0 = \begin{bmatrix} G(a_1, b_1, x_1) & \cdots & G(a_{\tilde{N}}, b_{\tilde{N}}, x_1) \\ \vdots & \ddots & \vdots \\ G(a_1, b_1, x_{N_0}) & \cdots & G(a_{\tilde{N}}, b_{\tilde{N}}, x_{N_0}) \end{bmatrix}_{N_0 \times \tilde{N}} \quad (9)$$

- (3) Calculating the initial output weight $\hat{\beta}^{(0)} = (H_0^T H_0)^{-1} H_0^T T_0$.

(4) Set $k = 0$, where k is the index representing the number of chunks of data presented to the network.

Sequential phase: given the $(k + 1)th$ chunk of new observations, $S_{k+1} = \{(x_i, t_i) | x_i \in R^n, t_i \in R^m, i = (\sum_{j=0}^k N_j) + 1, \dots, \sum_{j=0}^{k+1} N_j\}$, where N_{k+1} represents the number of newly obtained observations in the $(k + 1)th$ chunk.

- (1) Calculate the partial hidden layer output matrix h_{k+1}

$$h_{k+1} = \begin{bmatrix} G\left(a_1, b_1, x_{\left(\sum_{j=0}^k N_j\right)+1}\right) & \cdots & G\left(a_{\tilde{N}}, b_{\tilde{N}}, x_{\left(\sum_{j=0}^k N_j\right)+1}\right) \\ \vdots & \ddots & \vdots \\ G\left(a_1, b_1, x_{\left(\sum_{j=0}^{k+1} N_j\right)+1}\right) & \cdots & G\left(a_{\tilde{N}}, b_{\tilde{N}}, x_{\left(\sum_{j=0}^{k+1} N_j\right)+1}\right) \end{bmatrix}_{N_{k+1} \times \tilde{N}} \quad (10)$$

- (2) Calculate the output weight matrix β^{k+1}

$$\begin{cases} \hat{\beta}^{(k+1)} = \hat{\beta}^{(k)} + W_{k+1} (t_{k+1} - h_{k+1} \hat{\beta}^{(k)}) \\ W_{k+1} = \frac{P_k h_{k+1}^T}{I + h_{k+1} P_k h_{k+1}^T} \\ P_k = \left(I - \frac{P_{k-1} h_k^T}{I + h_k P_{k-1} h_k^T} \cdot h_k \right) P_{k-1} \end{cases} \quad (11)$$

- (3) Set $k = k + 1$. Then return to (1) in this sequential learning phase.

Here one can see that (9) has an intuitive physical meaning. The new output weight matrix $\hat{\beta}^{(k+1)}$ is equal to the last estimation $\hat{\beta}^{(k)}$ added with a correction term $W_{k+1} (t_{k+1} - h_{k+1} \hat{\beta}^{(k)})$, where $h_{k+1} \hat{\beta}^{(k)}$ represents the estimation or forecast value of t_{k+1} . That is to say, the foundation of the correction term is the difference between the actual $(k + 1)th$ output value t_{k+1} and its

estimation or forecast $h_{k+1}\hat{\beta}^{(k)}$ based on k th output weight estimation $\hat{\beta}^{(k)}$. And W_{k+1} is called weight matrix multiplied by the correction term.

Remark 1. It is necessary for $N_0 \geq \tilde{N}$ in the initialization phase of OS-ELM. Actually, we can remove this requirement after adding the regularized term in the calculation of output weights in the above analysis.

Remark 2. The modified term (9) of output weight matrix is equal to that in the OS-ELM algorithm despite the fact that the author employed two equations to update the output weights in Jian & Gao (2013). Here we just want to highlight the idea of weights, and then the correction term $P_{k+1}h_{k+1}^T(t_{k+1} - h_{k+1}\hat{\beta}^{(k)})$ in (Jian & Gao, 2013) is modified to $W_{k+1}(t_{k+1} - h_{k+1}\hat{\beta}^{(k)})$. A detailed derivation can refer to the following WOS-ELM algorithm.

PREDICTION MODEL FOR THE DESCENT OF BURDEN LAYER

The data in this study is obtained by the burden surface inspection system based on the industrial measurement radar developed by Chen et al. (2012.). Radar installation location and monitoring diagram are shown in Figures 2 and 3.

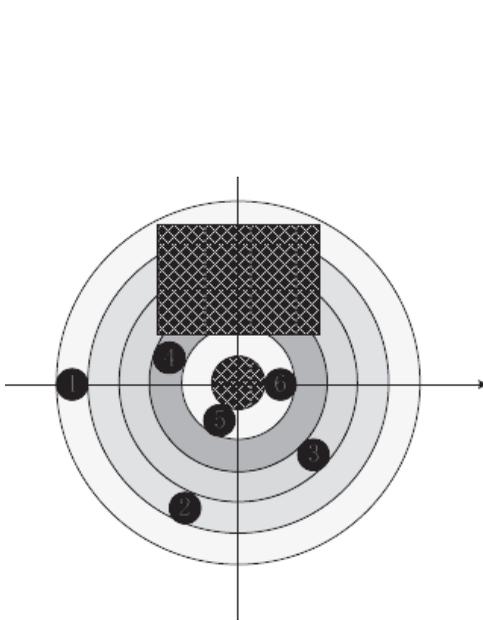


Figure 2. Radar installation diagram.

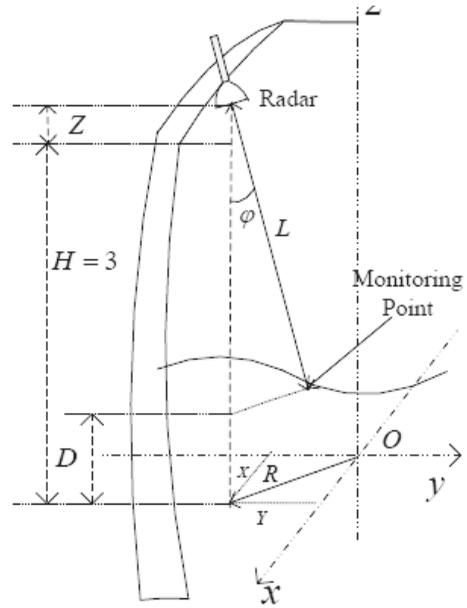


Figure 3. Radar detection coordinate.

Although the blast volume is large, there are many pieces of detection equipment installed on the top of BF, resulting in the fact that the installation area for the radar measurement system is limited (Wei & Chen, 2015). Figure 2 shows the installation locations of six radars. Considering the symmetrical feature of burden surface, we divide the surface into four loop regions (represented by four kinds of color). The radar cannot be installed in the shaded area. The four radars are mounted vertically above the four loop regions. Taking into account the importance of furnace center, the

remaining two radars are mounted aslant above the furnace center. Figure 3 presents the schematic diagram for a single radar monitoring. We have established a three-dimensional plot on the position of zero feed line. L is the distance from the radar installed position to the burden surface, which is also the initial detecting distance. R represents the distance from the projection point of radar installed position in the XOY plane to the coordinate origin. D is the vertical distance of the monitoring point. H represents the vertical distance from the throat of BF to the zero feed line, and $H = 3m$, while Z is the vertical distance between the radar installed position and the BF throat. Then one can calculate the radial and vertical distance of the monitoring point as follows:

$$\begin{cases} P = \sqrt{X^2 + Y^2} - L \sin(\varphi) \\ D = Z + H - L \cos(\varphi) \end{cases} \quad (12)$$

where P is the radial distance. And φ is the tilt angle of radar installation, which is known in advance.

Through the radar monitoring data, one can calculate the descent speed of burden layer as follows:

$$S_i = \frac{D_k - D_{k-1}}{\Delta t} \quad (13)$$

where S_i represents the descent speed monitored by the i th radar. D_k is the vertical distance at k th time, while Δt represents the sampling interval.

One can reconstruct the burden line based on radar detection data. BF is a symmetrical production vessel. The shape of burden line can reflect burden layer distribution. In addition, the iron-making process is a complex, nonlinear and strong coupling. Deep interactions exist among BF operating condition indexes. There are many indexes related to the descent speed of burden layer closely. In the speed prediction model, we combine the real radar data with the operating status of BF. Based on the operating experience, we chose the BF condition indexes related to the burden descent speed presented in Table 1.

Table 1. Blast furnace operating condition indexes.

Variable name	Unit
Blast volume	m ³ /min
Top temperature	°C
Top pressure	kPa
Pulverized coal injection	ton
Oxygen enrichment percentage	wt%
Blast pressure	kPa
Gas permeability	m ³ /min·kPa

The BF production is a continuous process. Both the single-step and multi-step prediction models are important to the operation of BF. We establish the descent speed prediction model from the following two aspects.

(1) Signal-step prediction model: ELM algorithm is employed in the descent speed prediction model as

$$S_i = ELM(S_{i-1}, S_{i-2}, BF \text{ Status Indexes}) \tag{14}$$

where we combine the last two descent speeds with the BF status indexes as the input variables. It is worth mentioning that the average descent speed is applied for the 5th and 6th radars above the center of BF.

(2) Multi-step prediction model: here the OS-ELM algorithm is employed in the multi-step speed prediction model as

$$S_i = ELM(\hat{S}_{i-1}, \hat{S}_{i-2}, BF \text{ Status Indexes}) \tag{15}$$

where \hat{S}_{i-1} and \hat{S}_{i-2} are the estimations of the last two descent speed (the output of ELM algorithm).

SIMULATION RESULTS

This section presents the simulation results of our proposed scheme for the descent speed prediction of burden layer. Here we collect the real radar monitoring data and the status indexes of 2500 m3 BF. Figure 4 represents the real radar monitoring data. The No. 3 radar has slight change during charging operation, which means that the burden monitored by No. 3 radar is hardly dropped. The data of No. 5 and No. 6 radars change significantly. Figure 5 presents the descent speed monitored by 6 industrial radars. One can clearly know that the burden around the center of BF has the fastest descent speed, which is 3 – 10 times more than the descent speed of burden in other locations. Within the first 5 minutes, the burden drops quickly and then becomes leveling off.

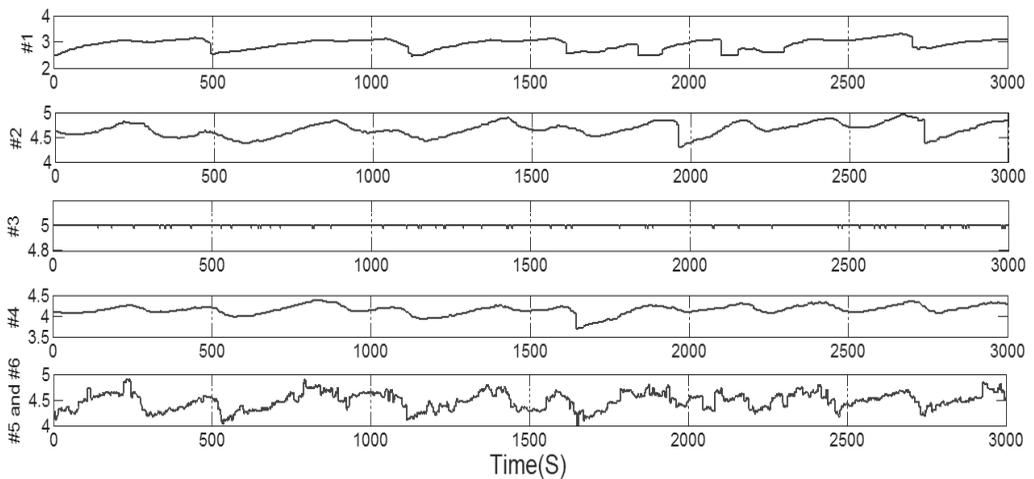


Figure 4. The real radar data.

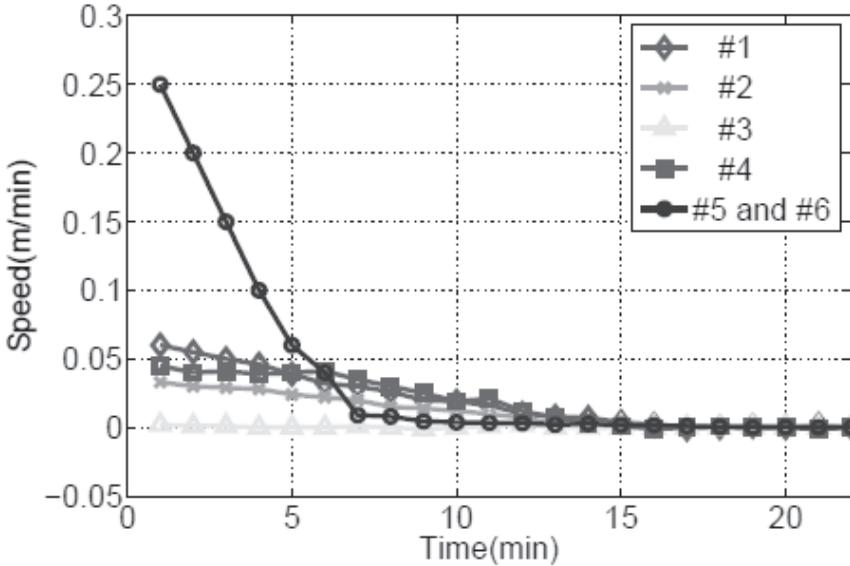


Figure 5. The descent speed.

The production environment of iron-making process is very harsh, resulting in the fact that the collected data often contain noise. It is necessary to make denoising implementation for the inputs of prediction model in the data preprocessing stage. The delay phenomenon in BF production is very common. We make correlation analysis between the prediction model inputs with the burden surface and determine the delay times.

Parameter selection

Theoretically, the model can be trained more and more accurately with the increase in hidden node numbers in ELM algorithm. However, it will take more training time. Considering the number of inputs in our model and the demand of online prediction, we select 100 hidden nodes in ELM and OS-ELM models. There are two types of activation functions taken into consideration.

(1) Sigmoidal additive activation function $G(a, b, x) = 1/(1 + \exp(-(a \cdot x + b)))$, where the input weights and biases are randomly generated from the range $[-1, 1]$.

(2) Gaussian RBF activation function $G(a, b, x) = \exp(-b\|x - a\|^2)$, where the centers are randomly generated from the range $[-1, 1]$, and the impact widths b are chosen from the range $[0, 1]$.

All the simulations have been conducted in Matlab 7.8.0 (2009a) running on a desktop PC with AMD Athlon(tm) II X2 250 processor, 3.00-GHz CPU, and 2G RAM. All the simulation results are averaged 50 times.

Single-step prediction results

Here we present the simulation results of single prediction in Figure 6. The blue line marked by squares represents the real descent speed. The red line with circles shows the ELM output with RBF active function, while the green marked by forks represents the model output using sigmoid

function. Taking into account the little change of the descent speed monitored by No. 3 radar, we omit it in this simulation.

From Figure 6, one can see that the output of ELM algorithm can well track the actual descent speed. In contrast, ELM algorithm with sigmoid nodes obtains more accurate and satisfied simulation results than those with RBF nodes. From Table 2, one can obtain the mean squares error (MSE) of ELM algorithm with sigmoid and RBF hidden nodes, which is the same as the above presentation.

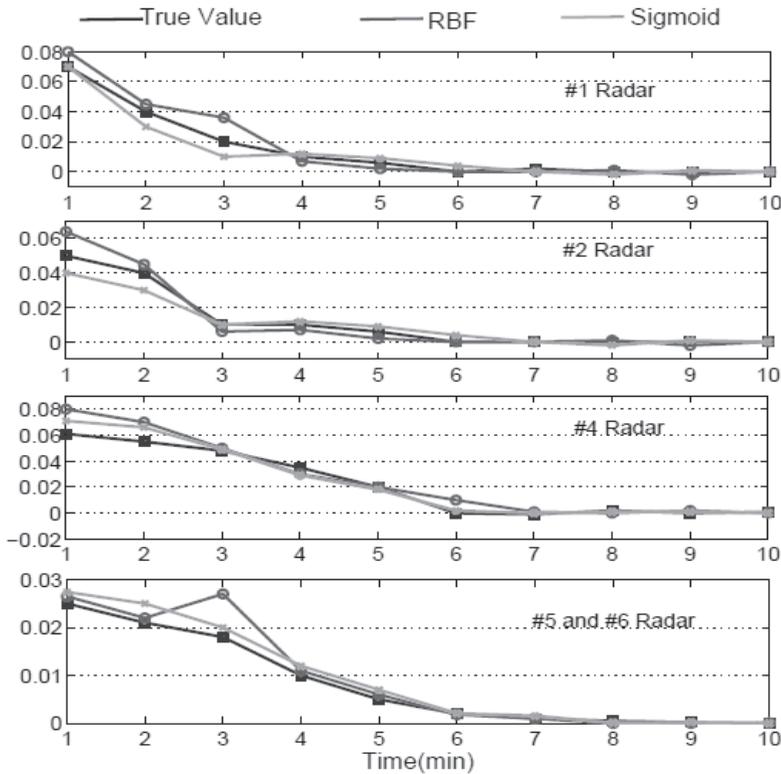


Figure 6. Simulation result of single-step prediction.

Multi-step prediction results

OS-ELM algorithm is employed in the multi-step prediction model, where the estimation output of OS-ELM model is used in the next prediction as input. For simplicity, the data collected from No. 5 and No. 6 radars is employed in this simulation. We make the data arrive one-by-one, and the initial number of chunks of training data is set as 300. Based on the simulation results of ELM algorithm, the sigmoid active function is applied in the multi-step prediction model. Figure 7 presents the multi-step prediction results, where one can see that OS-ELM algorithm can well track the change of descent speed in the first 7 steps (in the left side of green line). From Table 2, one can see the MSE of OS-ELM output, which is less than that of the ELM model.

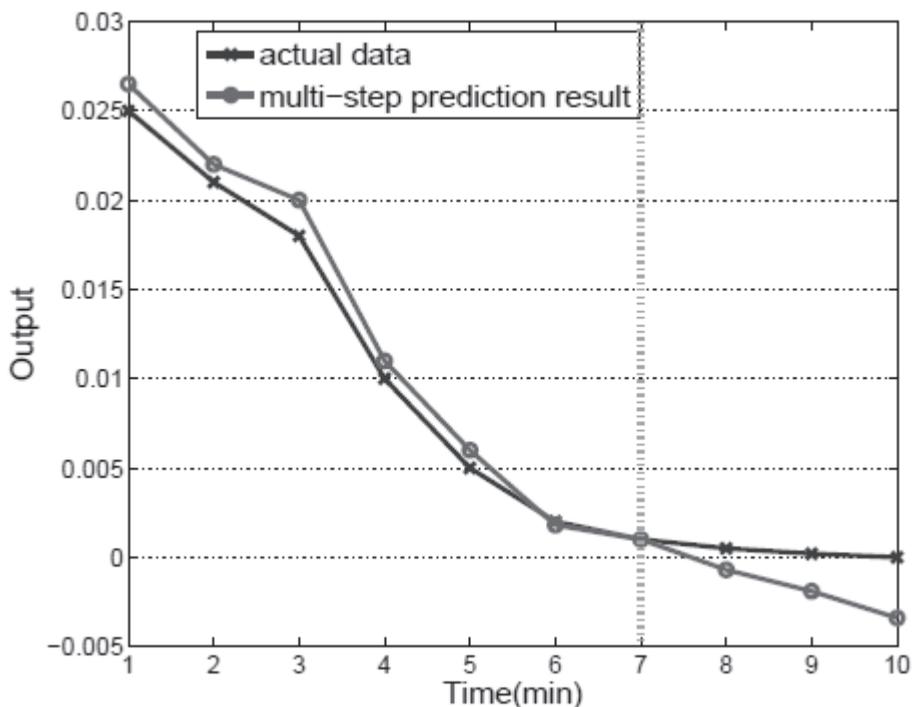


Figure 7. Simulation result of multi-step prediction.

Table 2. The comparison results.

Algorithm	Parameters	Training time	MSE
SVR	(C, γ) #SVs	0.1721	0.131
	(20,0.25)		
BP	#nodes	100	0.112
ELM-RBF		100	0.087
ELM-Sigmoid		100	0.080
OS-ELM-Sigmoid		100	0.072

Comparison results

In order to test the performance of ELM algorithm, we compare it with other famous algorithms, such as SVM and BP. The simulation results are presented in Table 2. ELM algorithm can obtain more satisfied and accurate results compared with SVM and BP methods with the least training time. The performance of OS-ELM algorithm is better than that of the ELM algorithm, despite a little more training time.

CONCLUSIONS

The descent speed of burden layer in BF is important for the operation of next charging. In this paper, we establish the single- and multi-step prediction model for the descent speed. The actual industrial radar monitoring data and the operating condition information are employed in the prediction model. In the simulation experiments, we collect the actual operating data from $2500m^3$ BF. The simulation results have verified the satisfied performance of the proposed scheme. In addition, ELM and OS-ELM algorithms are compared with SVM and BP algorithms, and they obtained more accurate results.

Acknowledgments

This work has been supported by the National Natural Science Foundation of China (NSFC Grant no. 61333002) and the National Natural Science Foundation of China (NSFC Grant no. 61673056).

REFERENCES

- C.H. Gao, L. Jian & S. H. Luo. 2012.** Modeling of the Thermal State Change of Blast Furnace Hearth With Support Vector Machines. *IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS*. **59**(2):1134-1145.
- G. B. Huang. 2015.** What are Extreme Learning Machines? Filling the Gap between Frank Rosenblatt's Dream and John von Neumann's Puzzle. *Cognitive Computation*. **7**(3): 263-278.
- G. B. Huang, D. H. Wang & Y. Lan. 2011.** Extreme learning machines: a survey. *International Journal of Machine Learning and Cybernetics*. **2**(2): 107 – 122.
- G. B. Huang, H. Zhou, X. Ding & R. Zhang. 2012.** Extreme learning machine for regression and multiclass classification. *IEEE Transactions on Systems Man & Cybernetics Society*. **42**(2): 513 – 529.
- G. B. Huang, Q. Y. Zhu & C. K. Siew. 2006.** Extreme learning machine: Theory and applications. *Neurocomputing*, **70**(1-3): 489 – 501.
- G. Feng, G. B. Huang, Q. Lin & R. Gay. 2009.** Error minimized extreme learning machine with growth of hidden nodes and incremental learning. *IEEE Transactions on Neural Networks*, **20**(8): 1352 – 1357.
- H. SAXE'N & J. HINNELA. 2010.** MODEL FOR BURDEN DISTRIBUTION TRACKING IN THE BLAST FURNACE. *Mineral Processing & Extractive Metallurgy Review*. **25**(1):1-27.
- J.D. Wei, X.Z. Chen, J. R. Kelly & Y.Z. Cui. 2015.** Blast Furnace Stockline Measurement using Radar. *Ironmaking & Steelmaking*, **42**(7): 533-541.
- L.M. Liu, A. Wang, M. Sha, X.Y. Sun & Y.L. Li. 2011.** Optional SVM for Fault Diagnosis of Blast Furnace with Imbalanced Data. *ISIJ International*, **51**(9): 1474-1479.
- L. Jian & C. H. Gao. 2013.** Binary Coding SVMs for the Multiclass Problem of Blast Furnace System. *IEEE Transactions on Industrial Electronics*, **60**(9):3846-3856.
- M. Ichida, T. Anan, M. Takao, K. Kakiuchi & Y. Morizane. 2007.** Inner profile and burden descent behavior in the blast furnace. *Tianjin Metallurgy*, **142**(4):46.
- N. Y. Liang, G. B. Huang, P. Saratchandran & N. Sundararajan. 2006.** A Fast and Accurate Online Sequential Learning Algorithm for Feedforward Networks. *IEEE Trans. Neural Networks*, **17**(6): 1411 – 1423
- R. S. NICK, A. TILLIANDER, T. L. I. JONSSON & P. G. JONSSON. 2013.** Mathematical Model of Solid Flow Behavior in a Real Dimension Blast Furnace. *ISIJ International*. **53**(6): 979-987.
- S.H. Luo, J. Huang, J. Zeng & Q. Zhang. 2011.** Identification of the Optimal Control Center for Blast Furnace Thermal State Based on the Fuzzy C-means Clustering. *ISIJ International*, **51**(10): 1668-1673.
- S. Scardapane, D. Comminiello, M. Scarpiniti & A. Uncini. 2015.** Online Sequential Extreme Learning Machine With Kernels. *IEEE Transaction on Neural Networks & Learning systems*, **26**(9):2212-2220.
- T. Mittal & R. K. Sharma. 2016.** An improved SVM using predator prey optimization and Hooke-Jeeves method for speech recognition. *Journal of Engg. Research*, **4**(1):1-20.
- X.Z Chen, J.D. Wei, D. Xu, Q. W. Hou & Z. Bai. 2012.** 3-Dimension Imaging System of Burden Surface with 6-radars Array in a Blast Furnace. *ISIJ International*, **52**(11): 2048-2054.

Submitted: 13/05/2016

Revised : 13/01/2017

Accepted : 17/01/2017