Available online at www.sciencedirect.com

ScienceDirect

Thermochimica Acta 454 (2007) 64–68

thermochimica acta

www.elsevier.com/locate/tca

Short communication

Predicting coal ash fusion temperature based on its chemical composition using ACO-BP neural network

Y.P. Liu*, M.G. Wu, J.X. Qian

Institute of Industrial Control Technology, College of Info Science and Engineering, Zhejiang University, Hangzhou 310027, China

Received 11 September 2006; received in revised form 24 October 2006; accepted 30 October 2006
Available online 12 November 2006

Abstract

Coal ash fusion temperature is important to boiler designers and operators of power plants. Fusion temperature is determined by the chemical composition of coal ash, however, their relationships are not precisely known. A novel neural network, ACO-BP neural network, is used to model coal ash fusion temperature based on its chemical composition. Ant colony optimization (ACO) is an ecological system algorithm, which draws its inspiration from the foraging behavior of real ants. A three-layer network is designed with 10 hidden nodes. The oxide contents consist of the inputs of the network and the fusion temperature is the output. Data on 80 typical Chinese coal ash samples were used for training and testing. Results show that ACO-BP neural network can obtain better performance compared with empirical formulas and BP neural network. The well-trained neural network can be used as a useful tool to predict coal ash fusion temperature according to the oxide contents of the coal ash.

Keywords: Coal ash fusion temperature; ACO-BP neural network; BP neural network; Chemical composition of coal ash

1. Introduction

Coal ash fusion temperature is important to boiler designers and operators. The slagging potential of coal affects the thermal efficiency and safe operation of steam boilers in coal-fired power stations [1,2]. Coal with a low ash fusion temperature is vulnerable to slag. Melting coal ash accumulates around heat transfer pipes and leads to corrosion of furnace components. Power stations have to shut down periodically to remove the clinker from heat transfer pipes. Softening temperature (ST) of coal ash correlates strongly to the slagging potential among the four fusion temperatures, deformation temperature (DT), ST, hemisphere temperature (HT) and flow temperature (FT); so the ash fusion temperature denotes ST in this paper.

Coal is a complex heterogeneous mixture, and the inorganic part transforms into ash during combustion [3]. Experimentally obtaining the fusion temperature is a time-consuming and troublesome job and needs special instruments. Another way to estimate ash fusion temperature is according to the chemical composition of coal ash, which is based on the fact that the chemical composition of coal ash determines its fusion temperature [4–9]. However, coal ash is a complex mixture of mineral matters, which mainly consists of SiO_2, Al_2O_3, Fe_2O_3, CaO, MgO, K_2O, Na_2O, TiO_2 and a few other oxides [2,10,11], and the ash fusion temperature regression formulas include a number of interacting factors whose relationship is not precisely known. The results calculated by the empirical formula usually do not have sufficient accuracy when the types of coal vary in a large range.

Artificial neural networks have powerful, nonlinear, mapping ability, and the modeling process is easier and more direct than for empirical models. It is not necessary to specify mathematical relationships between the input and output variables. The ACO-BP algorithm is a novel method to train neural networks. It overcomes the drawback in the back-propagation (BP) [12] algorithm to converge on local optima when the training surface is a multimodal distribution. This paper demonstrates a method of modeling coal ash fusion temperature with the ACO-BP neural network based on its chemical composition.

2. Methods

BP algorithm is currently the most widely used technique for training neural networks [13,14]. Suppose a set of P training

[*] Corresponding author. Tel.: +86 571 87951443; fax: +86 571 87951068.
E-mail address: ypliu@iipc.zju.edu.cn (Y.P. Liu).

0040-6031/$ – see front matter © 2007 Published by Elsevier B.V.
doi:10.1016/j.tca.2006.10.026

samples is available, the problem can be characterized as the process of minimizing the following sumsquared error:

$$J(W)=\frac{1}{2}\sum_{s=1}^{P}\sum_{i=1}^{N_M}(d_{s,i}-y_{s,i}^{M})^2 \quad (1)$$

where $d_{s,i}$ and $y_{s,i}^{M}$ are the ith target and actual outputs corresponding to the sth training pattern, W is a vector composed of all the weights and biases involved in the network, and N_M is the number of output units. In this scheme, an initial weight vector W_0 is iteratively adapted according to the following recursion to find an optimal weight vector. The positive constant η is the learning rate.

$$W_{k+1}=W_k-\eta\frac{\partial J(W)}{\partial W} \quad (2)$$

The BP algorithm can be described as follows. In the first phase, the actual outputs of the network are computed forward from the input layer to the output layer. While in the second phase, the descent gradient is calculated in a back-propagation fashion, which makes it possible to adjust the weights in a descent direction. This procedure is repeatedly performed for each training pattern until all error signals between the desired and actual outputs are sufficiently small. Obviously, BP algorithm is gradient descent in essence.

The ant colony optimization algorithm (ACO) draws its inspiration from the behavior of real ants as they move from their nest towards a food source [15,16]. ACO has been successfully applied to solve some complex combinatorial optimization problems with NP-hard characteristic, such as traveling salesman problems (TSP) [17], quadratic assignment problems [18], vehicle routing problems [19], and scheduling problems [20].

The basic idea of ACO-BP neural network can be depicted as follows. After the architecture of a neural network is selected, it needs to be trained before being used. Given ***D*** parameters in the network, which consist of all the weights and biases, the networks' evolution can be regarded as the process of searching for the optimal combination of the ***D*** parameters in their solution spaces. There are numerous candidate points for each parameter, so the candidate combination solutions are also numerous, and it is probable that the combinatorial optimization function has a multimodal distribution.

First, each parameter's definition spaces are split into a set of discrete points. Thus, each point is a candidate value of the corresponding parameter. As far as an ant is concerned, it can only can choose a value for each parameter among the candidate points, just like an ant visiting a city only once in solving the TSP. A pheromone table, Table 1, is needed for each parameter, where w_i is the ith parameter to be optimized, a_i denotes the divided calibration called a point, $\tau(i)$ represents the pheromone intensity of point a_i, and m is the number of shares that the space is divided into. So there are $(m+1)$ total points for each parameter.

Table 1
Pheromone table for each weight or bias

	w_i			
Tag	1	2	...	$m+1$
Split calibration	a_1	a_2	...	a_{m+1}
Pheromone intensity	$\tau(1)$	$\tau(2)$	...	$\tau(m+1)$

Several groups of better combinations of parameters can be provided by the ACO scheme. The BP algorithm initializes the weights of the network with these values and begins to train the network. Since ACO provides the BP with several groups of better initial values, the risk of being trapped in the local optima sharply decreases. Consequently, both the training effectiveness and evolving speed can be enhanced. The basic idea of the hybrid algorithm of ACO and BP is to use ACO to search several groups for better combinations of all the network parameters, and then use the BP algorithm to find the accurate value of each parameter. The detailed framework of the ACO-BP scheme is shown in Fig. 1. A complete description of the ACO-BP neural network is given in [21].

Data on 80 typical coal ash samples from different parts of China are studied. The types of the coal include brown coal, soft coal and anthracite coal, and their ranks range from low to high. The limits of oxides in the chemical composition of ash are shown in Table 2. The chemical composition of ash is in weight basis. The data were divided into two subsets, 60 of which were used for training and the remainder used for testing the network. To test the robustness and accuracy of the ACO-BP approach, the network was trained and tested with eight different combinations of the two subsets.

A three-layer feedforward neural network can approximate any nonlinear continuous function to an arbitrary accuracy [22], so this architecture was adopted in our network model. Because K_2O and Na_2O have similar chemical characteristics in the aspect of influencing the ash fusion temperature, their contents were combined as one input. Therefore, seven neurons were used in the input layer to denote the oxide contents. One neuron was adopted in the output layer to represent the fusion temperature. There are no general rules to determine the number of hidden nodes [23]. The common way is to set a relative large number at the beginning, and then reduce it to satisfy the error demand. By this method, the number of the hidden nodes is set as 10 finally. The transfer function of the hidden layer was the standard *sigmoid*, and the function of output was *purelin*. For the ACO-BP scheme, the number of ants, $W_{\min}$, $W_{\max}$ and the number of maximum iteration were selected as 43, −2, +2 and 200, respectively. In the best solution-keeping scheme, 10 better

Table 2
Minimum and maximum contents of the oxides

Oxides content	SiO_2 (%)	Al_2O_3 (%)	Fe_2O_3 (%)	CaO (%)	MgO (%)	K_2O+Na_2O (%)	TiO_2 (%)
Minium	15.17	3.66	1.53	0.16	0.03	0.01	0.46
Maximum	68.12	35.69	62.56	23.90	10.10	7.12	2.81

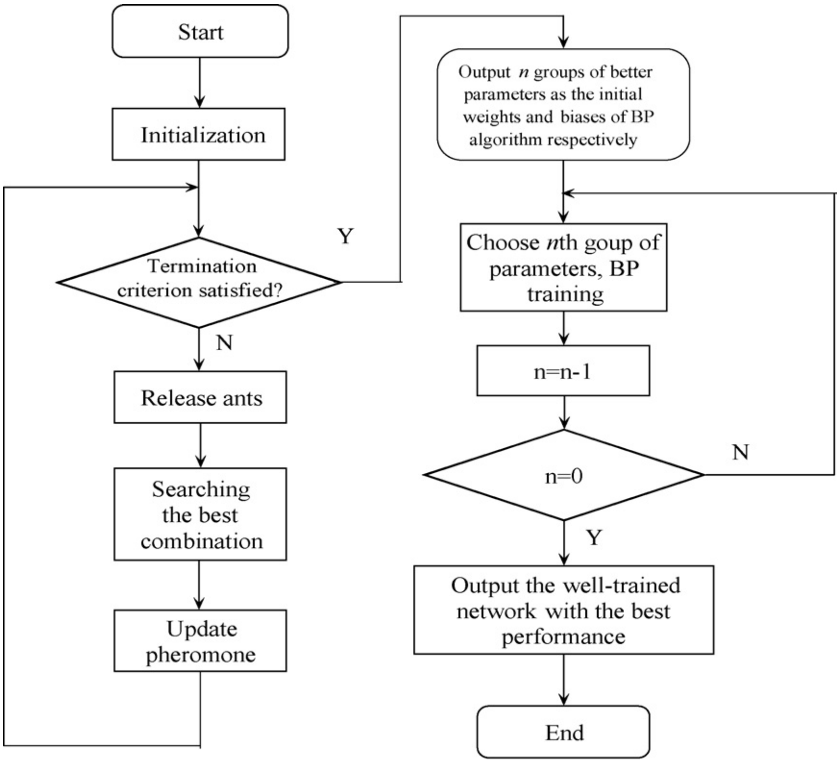
Start
Initialization
Termination criterion satisfied?
Y
N
Release ants
Searching the best combination
Update pheromone
Output n groups of better parameters as the initial weights and biases of BP algorithm respectively
Choose nth goup of parameters, BP training
n=n-1
n=0
N
Y
Output the well-trained network with the best performance
End

Fig. 1. Framework of training neural network using ACO-BP algorithm.

solutions were remembered. Both the BP in ACO-BP and the BP had the same parameter settings except that the iteration of the former was set as 10,000 times, and the later used 30,000 times. Each experiment was repeated 10 times independently.

3. Results and discussion

3.1. Comparison of ACO-BP and BP algorithms

Comparisons of output results using the ACO-BP and BP neural networks are shown in Table 3, where *training error* and *predicting error* are calculated by the following equation:

$$\mathrm{Std}\,(x_i) = \frac{1}{M}\sum_{i=1}^{M}\frac{|\hat{y}(x_i) - y_i|}{y_i} \qquad (3)$$

where $\hat{y}(x_i)$ represents the neural network output when the input is x_i, y_i denotes the actual fusion temperature, and M is the number of outputs. The results shown in Table 3 are the average of 10 experiments.

The maximum training error of the ACO-BP algorithm is 1.78%, the minimum error is 1.39% and the average training error is 1.55%. All are less than the corresponding training errors of BP. More important, the ACO-BP obtained smaller predictive errors, 5.98% compared to 5.16%.

A group of randomly chosen training and testing results are shown in Fig. 2. The ACO-BP training and testing samples are uniformly distributed near the line of 45°, which indicates that the outputs of the network accord well with the actual target values. Comparatively, the BP algorithm does not attain this goal; the training samples are relatively uniformly distributed near the line of 45°, but some of the testing samples are far

Table 3
Comparison of ACO-BP and BP neural networks

No.	ACO-BP		BP	
	Training error (%)	Predicting error (%)	Training error (%)	Predicting error (%)
1	1.50	3.81	1.76	4.44
2	1.44	6.73	1.99	8.87
3	1.53	6.97	1.83	8.07
4	1.39	6.41	1.78	7.75
5	1.78	4.02	2.02	4.11
6	1.55	4.17	1.78	4.71
7	1.47	4.72	1.75	5.30
8	1.76	4.41	1.93	4.57
Average	1.55	5.16	1.85	5.98

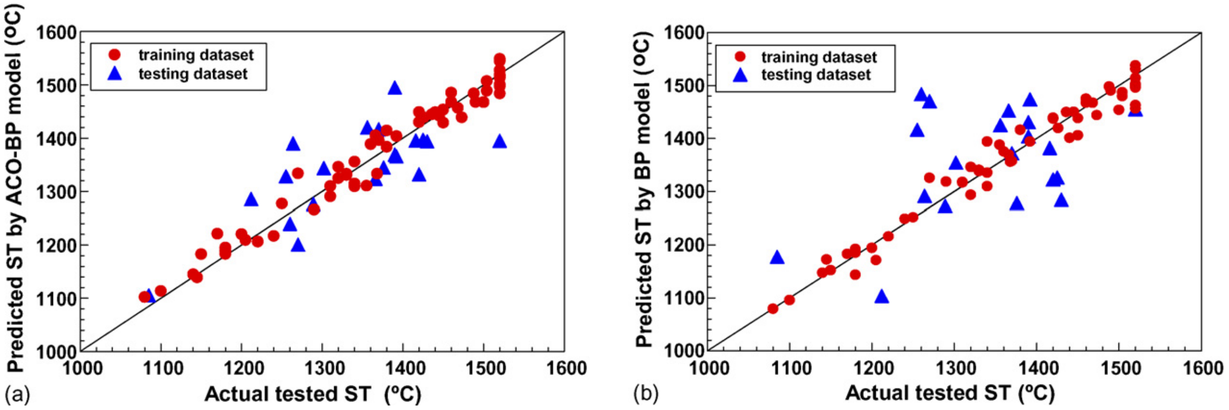

Fig. 2. The performance comparison of ACO-BP and BP neural networks.

away from this line. The main reason lies in the fact that ACO-BP has a higher probability of escaping the local optima of the error surface than BP algorithm.

3.2. Comparison of ACO-BP and empirical formulas

The regression formulas usually used for predicting coal ash fusion temperature in China [24] are:

(1) When the content of the SiO_2 is no larger than 60% and the content of Al_2O_3 is larger than 30%, the fusion temperature is calculated by

$$T = 69.94{\cdot}SiO_2 + 71.01{\cdot}Al_2O_3 + 65.23{\cdot}Fe_2O_3 + 12.16{\cdot}CaO + 68.31{\cdot}MgO + 67.19{\cdot}a - 5485.7 \tag{4}$$

(2) When the content of the SiO_2 is no larger than 60%, the content of Al_2O_3 is no larger than 30% and the content of Fe_2O_3 is no larger than 15%, the fusion temperature is calculated by

$$T = 92.55{\cdot}SiO_2 + 97.83{\cdot}Al_2O_3 + 84.52{\cdot}Fe_2O_3 + 83.67{\cdot}CaO + 81.04{\cdot}MgO + 91.92{\cdot}a - 7891 \tag{5}$$

(3) When the content of the SiO_2 is no larger than 60%, the content of Al_2O_3 is no larger than 30% and the content of Fe_2O_3 is larger than 15%, the fusion temperature is calculated by

$$T = 1531 - 3.01{\cdot}SiO_2 + 5.08{\cdot}Al_2O_3 - 8.02{\cdot}Fe_2O_3 - 9.69{\cdot}CaO - 5.861{\cdot}MgO - 3.99{\cdot}a \tag{6}$$

(4) When the content of the SiO_2 is larger than 60%, the fusion temperature is calculated by

$$T = 10.75{\cdot}SiO_2 + 13.03{\cdot}Al_2O_3 - 5.28{\cdot}Fe_2O_3 - 5.88{\cdot}CaO - 10.28{\cdot}MgO + 3.75{\cdot}a + 453 \tag{7}$$

where $a = 100 - (SiO_2 + Al_2O_3 + Fe_2O_3 + CaO + MgO)$.

When the same data were used to test the predictive performance of the regression formulas, the maximum error is as much as 61.54%, and the average error is 8.54%, both of which are larger than those of the ACO-BP and BP neural networks. The main reason lies in the fact that the empirical formulas are only a rough approximator of nonlinear relationships between the chemical composition of coal ash and its fusion temperature based on current samples. However, the real relationships are more complex. Therefore, the empirical formulas will fail in some samples. The predictive results based on the formula are shown in Fig. 3.

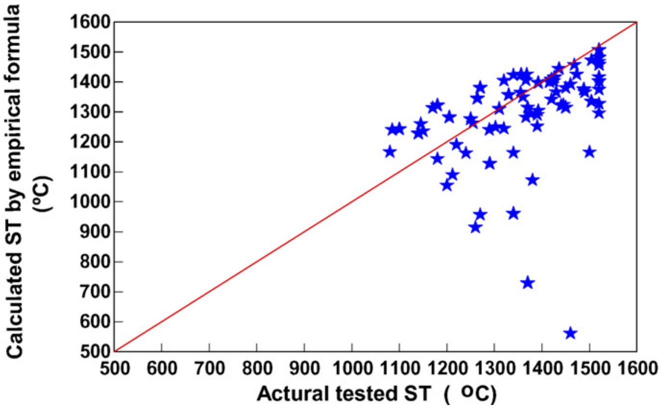

Fig. 3. Results based on empirical formulas.

The sample data of coal ash are typically different types and ranks from different parts of China, so the well-trained neural network can work well even when the ranks and types of coal change a lot. Furthermore, the artificial neural networks have the powerful learning ability. If a new type or rank of coal does not accord well with the well-trained network model, the ACO-BP algorithm can be easily retrained to adjust the weights of the neural network to adapt it. In contrast, revising the empirical formula will be a troublesome procedure.

Acknowledgements

The authors gratefully acknowledge Prof. J.P. Jiang and two anonymous referees for their helpful comments on drafts of this paper.

References

[1] S.A. Lolja, H. Haxhi, R. Dhimitri, S. Drushku, A. Malja, Fuel 81 (2002) 2257–2261.

[2] C.G. Yin, Z.Y. Luo, M.J. Ni, K.F. Cen, Fuel 77 (1998) 1777–1782.

[3] L.A. Hansen, F.J. Frandsen, K. Dam-Johansen, S.S. Henning, Thermochim. Acta 326 (1999) 105–117.

[4] H. Kahraman, A.P. Reifenstein, C.D.A. Coin, Fuel 78 (1999) 1463–1471.

[5] D. Thompson, B.B. Argent, Fuel 78 (1999) 1679–1689.

[6] M. Zevenhoven-Onderwater, J.-P. Blomquist, B.-J. Skrifvars, Fuel 79 (2000) 1353–1361.

[7] L.Y. Huang, J.S. Norman, M. Pourkashanian, A. Williams, Fuel 75 (1996) 271–279.

[8] V.R. Gray, Fuel 66 (1987) 1230–1239.

[9] S.V. Vassilev, K. Kitano, S. Takeda, T. Tsurue, Fuel Process. Technol. 45 (1995) 27–51.

[10] E. Jak, Fuel 81 (2002) 1655–1668.

[11] H. Zhou, L.G. Zheng, J.R. Fan, K.F. Cen, J. Zhejiang University 38 (2004) 1479–1482.

[12] D.E. Rumelhart, G.E. Hinton, R.J. Williams, Nature 323 (1986) 533–536.

[13] R.S. Sexton, R.E. Dorsey, Decis. Support Syst. 30 (2000) 11–22.

[14] Z.J. Liu, A.X. Liu, C.Y. Wang, Z. Niu, Future Gener. Comput. Syst. 20 (2004) 1119–1129.

[15] A. Colorni, M. Dorigo, V. Maniezzo, Proceedings of First European Conference on Artificial Life, 1991.

[16] M. Dorigo, C. Blum, Theor. Comput. Sci. 344 (2005) 243–278.

[17] M. Dorigo, L.M. Gambardella, IEEE Trans. Evol. Comput. 1 (1997) 53–66.

[18] V. Maniezzo, A. Colorni, IEEE Trans. Knowl. Data Eng. 5 (1999) 769–778.

[19] J.E. Bell, R.M. Patrick, Adv. Eng. Inform. 18 (2004) 41–48.

[20] C. Blum, M. Sampels, J. Math. Model. Algorithm 3 (2004) 285–308.

[21] Y.P. Liu, M.G. Wu, J.X. Qian, LNCS 3971 (2006) 714–722.

[22] K. Hornik, M. Stinchcombe, H. White, Neural Networks 2 (1989) 359–366.

[23] L. Wang, Tsinghua University Press, 2001.

[24] W.M. Chen, N. Jiang, Clean Coal Technol. 2 (1996) 34–37.