Hindawi Publishing Corporation
Scientific Programming
Volume 2015, Article ID 146410, 11 pages
http://dx.doi.org/10.1155/2015/146410

Research Article

Hybrid Multiple Soft-Sensor Models of Grinding Granularity Based on Cuckoo Searching Algorithm and Hysteresis Switching Strategy

Jie-Sheng Wang[1] and Na-Na Shen[2]

[1]*National Financial Security and System Equipment Engineering Research Center, University of Science & Technology Liaoning, Anshan 114044, China*
[2]*School of Electronic and Information Engineering, University of Science & Technology Liaoning, Anshan 114044, China*

Correspondence should be addressed to Jie-Sheng Wang; wang_jiesheng@126.com

Received 21 July 2015; Revised 25 November 2015; Accepted 25 November 2015

Academic Editor: Frédéric Lardeux

According to the characteristics of grinding process and accuracy requirements of technical indicators, a hybrid multiple soft-sensor modeling method of grinding granularity is proposed based on cuckoo searching (CS) algorithm and hysteresis switching (HS) strategy. Firstly, a mechanism soft-sensor model of grinding granularity is deduced based on the technique characteristics and a lot of experimental data of grinding process. Meanwhile, the BP neural network soft-sensor model and wavelet neural network (WNN) soft-sensor model are set up. Then, the hybrid multiple soft-sensor model based on the hysteresis switching strategy is realized. That is to say, the optimum model is selected as the current predictive model according to the switching performance index at each sampling instant. Finally the cuckoo searching algorithm is adopted to optimize the performance parameters of hysteresis switching strategy. Simulation results show that the proposed model has better generalization results and prediction precision, which can satisfy the real-time control requirements of grinding classification process.

1. Introduction

The grinding process is the main production process for the mineral concentrator factories, whose technique is complex and which is affected by many factors in the main loop, such as the varied ore characteristic, the ore hardness, the particle granularity distribution, the mineral composition, or the varied flow rate. Serious nonlinear, strong coupling and big time lag are the characteristics of the grinding process. Due to the restriction of on-site and the lack of mature detection devices, it is difficult to obtain the internal parameters (grinding granularity and milling ore ratio) of the grinding process in time, which resulted in not achieving the direct closed-loop control. The soft-sensor modeling technology can effectively solve the estimation of the industrial process quality indexes online [1].

In order to achieve the forecasting and monitoring for grinding granularity and milling ore ratio on time, the soft-sensor model is established by adopting the instrumental variables measured directly in grinding process. It has very important significance for the stability of the grinding process. In view of the grinding process, the domestic scholars have proposed many soft-sensor modeling methods based on neural network [2–4] and case-based reasoning [5]. Combining with the actual working conditions of the grinding classification process, a soft-sensor model is proposed based on the RBF neural network [2]. According to the characteristics of two-stage grinding process, a neural network soft-sensor model for grinding granularity is set up based on the multiple input layers neural network optimized by genetic algorithm (GA) [3]. Based on the idea that multiple models can improve the prediction accuracy and robustness, a multiple neural networks soft-sensor model of grinding granularity is proposed [4]. The case-based reasoning (CBR) technology is applied in the grind size prediction of grinding process [5]. Now a single model structure is most used in nonlinear soft-sensor model. In theory, if there is no limit on the model size and there are plenty of training data, the soft-sensor model

Table 1: Prediction data set of soft sensor model.

Number	Feeding capacity	Inlet water flow (m^3/s)	Export water flow (m^3/s)	Pump pool level (m)	Pump pressure (Kpa)	Current (A)	Feed flow of cyclone (m^3/s)	First concentration	Ball mill power (kw)	TP	Granularity (%)
1	137.39	28.75	86.65	1.61	0.042	62	−129.69	65.02	1126	390.86	58.54
2	136.37	28.92	95.64	1.58	0.041	64	−128.86	65.26	1155	398.62	56.16
3	137.40	28.79	84.87	1.59	0.042	64	−128.92	66.26	1153	435.55	57.99
4	136.06	28.84	100.47	1.58	0.042	62	−129.55	64.75	1120	375.86	57.06
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
600	136.53	28.98	63.93	1.69	0.042	68	−128.81	65.77	1204	279.12	58.51

based on neural network or fuzzy system can always obtain a satisfactory model structure and a predictive accuracy. But with the enlargement of the training area and the increased complexity of the systemic state sharply, the prediction accuracy, robustness, and generalization ability of the soft-sensor model are greatly reduced. The idea of multimodel switching can satisfy the requirement of complex working conditions. A predictive control model of superheated steam temperature for coal-fired power plants is proposed [6, 7]. The multiple model control strategy is applied in the blue tail ticket tracker (BTT) missile design [8].

A hybrid multiple soft-sensor model based on the cuckoo searching algorithm and the hysteresis switching strategy is proposed to predict the grinding granularity, which includes the mechanism soft-sensor model, the BP neural network soft-sensor model, and wavelet neural network soft-sensor model. At each sampling instant, the optimal local model is selected as the current soft-sensor model through CS-hysteresis switching strategy. The simulation results show that the method can significantly improve the accuracy and robustness.

2. Technological Flowchart of Grinding Process

The technique flowchart of a typical grinding process is shown in Figure 1 [9]. The grinding process is the following technique step after ore crushing process, whose purpose is to make all or most of useful ores reach monomer separation and avoid overgrinding on the same time. The typical two-stage closed-circuit grinding process is mainly composed of ball milling machine, spiral classifier, and hydrocyclone, where the first closed-circuit grinding process is composed of the first-stage ball milling machine and spiral classifier and the second closed-circuit grinding process is composed of the second ball milling machine and hydrocyclone.

The specific technique flowchart of the grinding classification process is described as follows. Ore grains are fed into the conveyer by the pendulum feeder and conveyed to ball milling machine for grinding. The rowing ore grains from ball milling machine go into the spiral classifier for the first grading. The coarse ore grains are returned to the first-stage ball milling machine by the conveyer for regrinding and the fine ore grains from the overflow inlet of cyclone go into the sand pump pool. Then the fine ore grains will be pumped into the hydrocyclone by the water pump for the secondary classification. By the centrifugal force of hydrocyclone, the different ore grains are divided from each other. The rather finer ore grains overflowing from the overflow outlet of the hydrocyclone will go into the next operation process. The coarser particles will go from the bottom flow outlet of the hydrocyclone into the secondary ball milling machine for regrinding. Thus these steps form a grinding closed loop. The grinding classification process is a complex controlled object. There are many factors influencing the key economic and technical indicators (grinding granularity), such as the milling feeding capacity, the inlet water flow, the export water flow, and the pump pool level. This paper adopts 600 groups' production data to establish soft-sensor model of grinding granularity, which is shown in Table 1.

3. Soft-Sensor Models of Grinding Granularity

3.1. Mechanism Model

3.1.1. Separation Granularity Model. Separation granularity is the grinding granularity from grit mouth and overflow mouth of hydrocyclone, each accounting for 50%. It is usually represented as d_{50}. According to the empirical model published by Splitter in 1976, the separation granularity model is described as follows:

$$d_{50(c)} = \frac{14.2 D_c^{0.46} D_i^{0.6} D_o^{1.21} \exp(0.063V)}{D_u^{0.71} h^{0.38} Q^{0.45} (S-L)^{0.5}}, \quad (1)$$

where $d_{50(c)}$ is the separation granularity of the hydrocyclone; D_c, D_o, and D_u are hydrocyclone feeding concentration, inner diameter of hydrocyclone overflow mouth, and inner diameter of hydrocyclone grit mouth, respectively; h is the distance between hydrocyclone overflow mouth and hydrocyclone grit mouth; V is the content of solid in hydrocyclone feeding pulp; Q is hydrocyclone feeding flow rate; S is the solid density of hydrocyclone feeding pulp; L is the density of hydrocyclone feeding pulp; P is the hydrocyclone pressure drop.

There is the following relationship between Q and P:

$$Q = 9.4 \times 10^{-3} \sqrt{P} D_c^2. \quad (2)$$

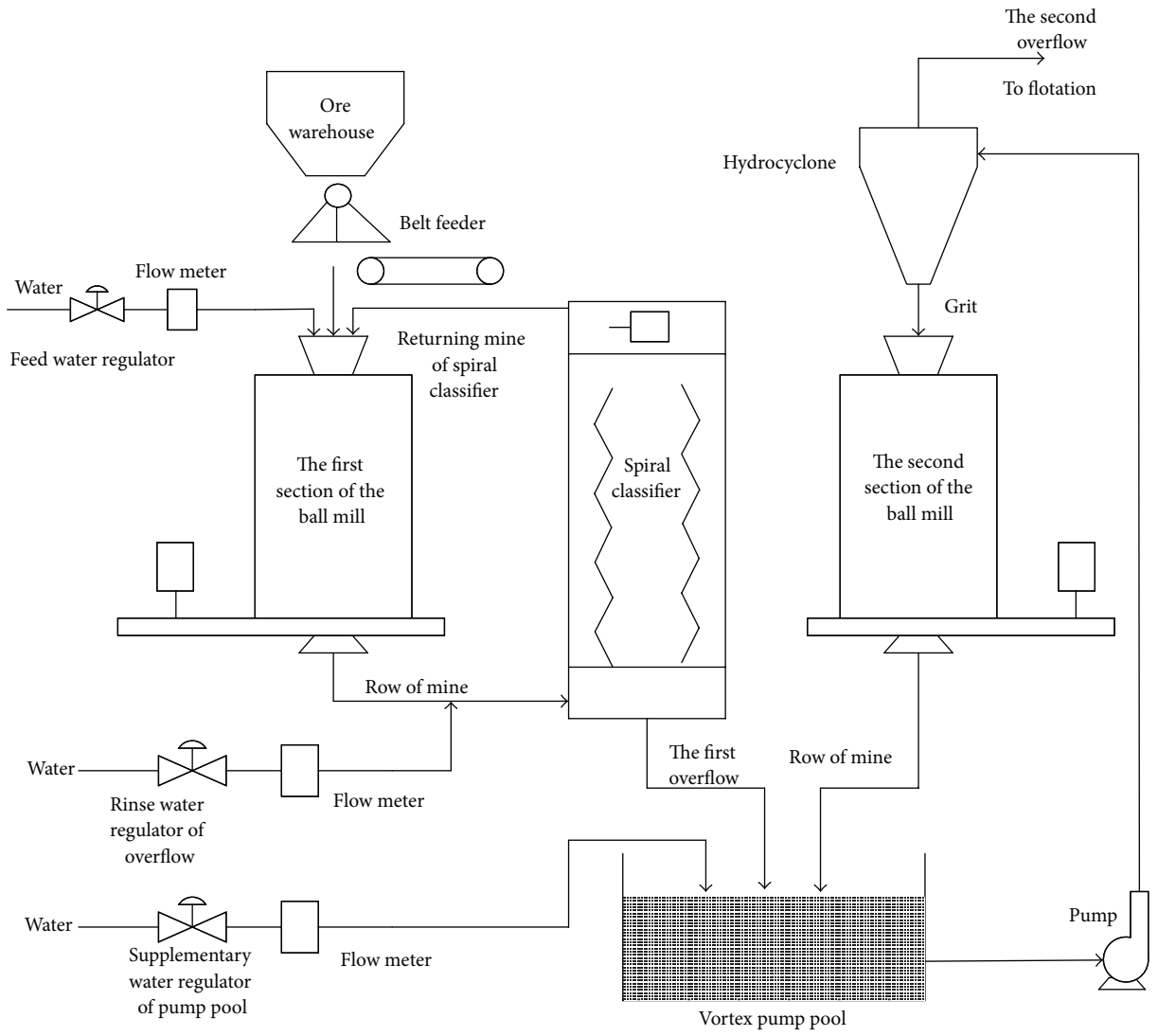

Figure 1: Technique flowchart of grinding process.

According to (1) and (2), the relationship between $d_{50(c)}$ and P is described as follows:

$$d_{50(c)} = \frac{14.2 D_c^{0.46} D_i^{0.6} D_o^{1.21} \exp(0.063V)}{D_u^{0.71} h^{0.38} Q^{0.45} (S-L)^{0.5}}$$
$$= \frac{14.2 D_c^{0.46} D_i^{0.6} D_o^{1.21}}{D_u^{0.71} h^{0.38} \left(9.4 \times 10^{-3} D_c^2\right)^{0.45} (S-L)^{0.5}} \quad (3)$$
$$\cdot \exp(0.063V) P^{-9/40} = \Theta \cdot \exp(0.063V) P^{-9/40},$$

where Θ is a variable associated with hydrocyclone structure parameters, which has no relationship between V and P.

Equation (3) is linearized for the convenience calculation as

$$\lg d_{50(c)} = \lambda_0 + \lambda_1 D_o + \lambda_2 D_i + \lambda_3 D_u + \lambda_4 Q + \lambda_5 D_c, \quad (4)$$

where λ_0, λ_1, λ_2, λ_3, λ_4, and λ_5 are the undetermined coefficients.

3.1.2. Theoretical Model of Grinding Granularity. Grinding granularity is referred to granularity range or the content of some specific granularity. The theory model of grinding granularity is described as follows [10]:

$$M_{200} = \frac{\sum_{i=1}^{d_{75}} m_i \left(1 - R_f - E_i \left(1 - R_f\right)\right)}{\sum_{i=1}^{N} m_i \left(1 - R_f - E_i \left(1 - R_f\right)\right)}, \quad (5)$$

where M_{200} is the quality percentage of 200 mesh (75 μm) mineral granularity in the whole classification products; d_{75} is the granularity size 75 μm; N is the biggest granularity size in classification products; m_i is the quality of the ith grade grinding granularity determined by the granularity distribution of hydrocyclone feeding pulp; R_f is the mass fraction of hydrocyclone bottom mouth, which has relations with water content of spinning pulp and structural parameters of hydrocyclone; E_i is the classification efficiency of the first i grade mineral granularity, which is decided by the structural parameters of hydrocyclone and operating parameters.

3.1.3. Relationship between Separation Model and Theoretical Model of Grinding Granularity. Most of grinding granularity distribution characteristics conform to the Rosin-Rammler granularity equation. So, the grinding granularity distribution $F_f(d)$ is represented as follows:

$$F_f(d) = 1 - \exp\left(-0.6931\left(\frac{d}{d_f(50)}\right)^k\right), \quad (6)$$

where $d_f(50)$ is the grinding granularity when hydrocyclone cumulative production rate is 50% and k is a constant related to the pulp properties.

According to conversion efficiency curve equation put forward by Plitt, E_i is calculated by

$$E_i = 1 - \exp\left(-0.6931\left(\frac{d_i}{d_{50(c)}}\right)^m\right), \quad (7)$$

where d_i is a diameter of the ith grade mineral granularity; m is related to the pulp and characteristics of grinding classification circuit.

Through comparisons of (5), (6), and (7), there is relationship between grinding granularity and separation granularity despite of the different concepts. This relationship function between M_{200} and $d_{50(c)}$ is described as follows [11]:

$$\lg(M_{200}) = k_0 + k_1(\lg d_{50(c)}), \quad (8)$$

where k_0, k_1 are undetermined coefficients.

Put (4) into (8), and regard properties of spinning pulp and structural parameters of the hydrocyclone as constants. So the mechanism model of grinding granularity is described as follows:

$$\lg(M_{200}) = a_0 + a_1 D_c + a_2 Q, \quad (9)$$

where a_0, a_1, and a_2 are undetermined coefficients.

According to (9), the grinding granularity can be expressed by hydrocyclone feeding concentration and hydrocyclone feeding flow rate. a_0, a_1, and a_2 are decided by the least squares method. So the grinding granularity is estimated online through (9) after the coefficients are determined.

These models are derived based on the ideal working conditions of hydrocyclone and a lot of experimental data of grinding process. But the grinding process is complex and time-varying, so these models do not have good practical application value. However these mathematical models provide the technical guidance for using soft-sensor technology to estimate the grinding granularity.

3.2. BP Neural Network. BP neural network is a kind of multiple layers feed-forward neural network, whose structure is shown in Figure 2.

In Figure 2, X_j represents the input of the input layer at node j, $j = 1, \ldots, m$; w_{ij} is the weight between node i in hidden layer and node j in input layer; θ_i is a threshold value of the ith hidden layer node; $\phi(x)$ is the excitation function of hidden layer; w_{ki} is a weight between node k in output layer and node i in hidden layer, $i = 1, \ldots, q$; a_k is a threshold value

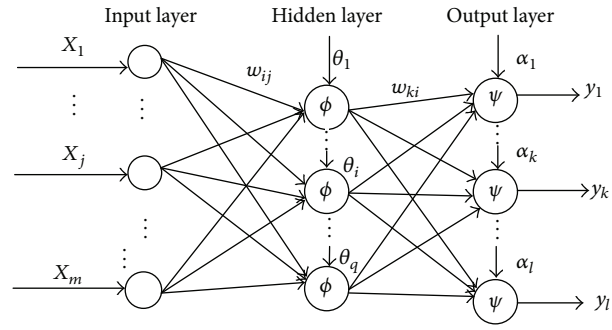

Figure 2: Structure of BP neural network.

of the kth output layer node, $k = 1, \ldots, l$; $\psi(x)$ is the excitation function of output layer; y_k is the output of the output layer at node k, $k = 1, \ldots, l$.

Back propagation (BP) algorithm is essentially a gradient descent method. The training of BP neural network can be seen as a process of searching minimal point for a multivariate function. Its basic idea is described as follows.

Step 1. Initialize each weight value to a small random number with distributed uniformly random numbers as the initial connection weights and the threshold values of the nodes.

Step 2. Calculate the actual output of BPNN:

(1) For the input layer nodes, their output O_j^I are equal to the input data X_j; that is to say; $O_j^I = X_j$, $j = 1, \ldots, m$.

(2) For the hidden layer nodes, their input is described as follows:

$$\text{net}_i^H = \sum_{j=1}^{m} w_{ij}^{HI} O_j^I, \quad i = 1, \ldots, q. \quad (10)$$

The output is

$$O_i^H = f\left(\text{net}_i^H - \theta_i^H\right), \quad (11)$$

where w_{ij}^{HI} is the connection weights between node i in hidden layer and node j in input layer; θ_i is a threshold value of hidden layer node i; q is the number of hidden layer nodes; O_j^I is the output of the input layer at node j, that is, X_j; f is Sigmoid function.

(3) Input of the output layer nodes is described as follows:

$$\text{net}_k^O = \sum_{i=1}^{q} w_{ki}^{OH} O_i^H, \quad k = 1, \ldots, l. \quad (12)$$

The output of the output layer nodes is

$$y_k = f\left(\text{net}_k^O - \theta_k^O\right), \quad (13)$$

where w_{ki}^{OH} is the connection weights between output layer node k and hidden layer node i; θ_k^O is a threshold value of the output layer node k.

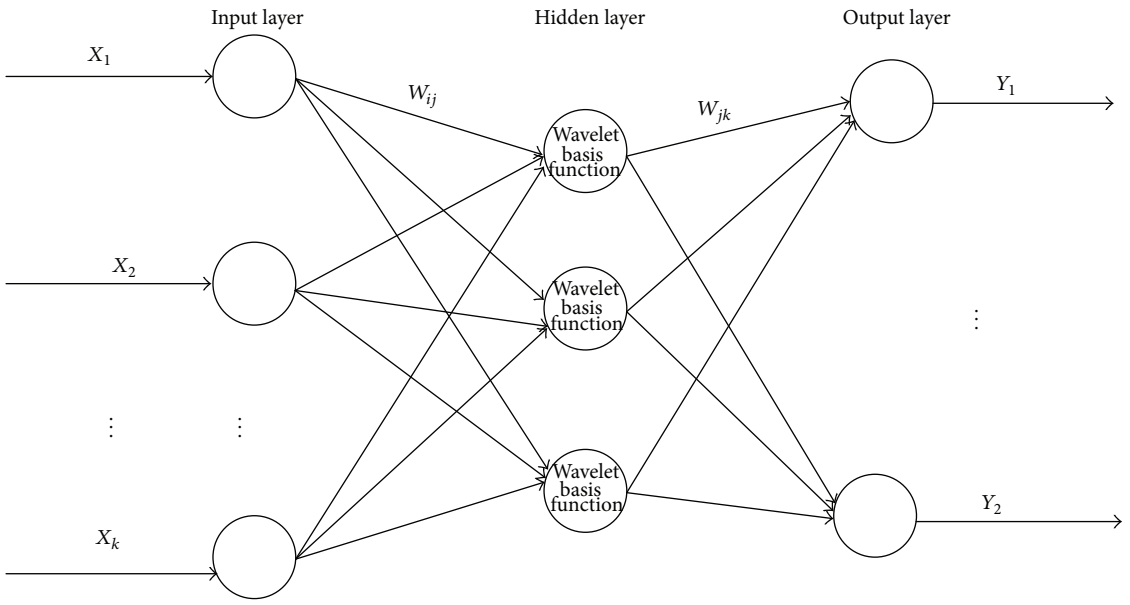

Figure 3: Structure of wavelet neural network.

Step 3. The error of the output node is calculated by the following equation:

$$e_k = d_k - y_k. \tag{14}$$

Then calculate the error squared sum of all output nodes and obtain the energy function:

$$E = \frac{1}{2}\sum_{k=1}^{l}\left(d_k - y_k\right)^2. \tag{15}$$

If E is less than predetermined value, turn to Step 5; otherwise continue to Step 4.

Step 4. Adjust the weights of BPNN.

(1) The weights between the output layer nodes and the hidden layer nodes are adjusted as follows:

$$\begin{aligned} \overline{w}_{ki}^{OH} &= w_{ki}^{OH} + \Delta w_{ki}^{OH}, \\ \Delta w_{ki}^{OH} &= \eta\sigma_k^O \cdot O_i^H, \\ \sigma_k^O &= \left(d_k - y_k\right)\cdot y_k\left(1 - y_k\right), \end{aligned} \tag{16}$$

where η is the training rate and general $\eta = 0.01$~1.

(2) The weights w_{ij}^{HI} between the hidden layer nodes and the input layer nodes are adjusted as follows:

$$\begin{aligned} \overline{w}_{ij}^{HI} &= w_{ij}^{HI} + \Delta w_{ij}^{HI}, \\ \Delta w_{ij}^{HI} &= \eta\sigma_i^H \cdot O_j^I, \\ \sigma_i^H &= O_i^H\left(1 - O_i^H\right)\sum_{k=1}^{l}\sigma_k^O w_{ki}^{OH}. \end{aligned} \tag{17}$$

Step 5. Carry on the next training samples. The learning process of BPNN is complete until each training sample satisfies the target.

In this paper, the multiple input and single output three-layer BP neural network is used. The topology of BP neural network is 10-20-1. The neuron transfer function in hidden layer used bipolar S type Tangent function (tansig):

$$f(x) = \frac{1 - e^{-x}}{1 + e^{-x}}, \quad (-1, 1). \tag{18}$$

The neuron transfer function in output layer uses the linear transfer function (purelin):

$$f(x) = x. \tag{19}$$

3.3. Wavelet Neural Network. The structure of wavelet neural network is similar to BP neural network; that is to say, the signal spreads forward while errors spread back. But the transfer function in hidden layer of wavelet neural network is the wavelet basis function [12], whose structure is shown in Figure 3.

In Figure 3, $X_1, X_2, \ldots, X_k$ are the inputs of wavelet neural network, $Y_1, Y_2, \ldots, Y_k$ are the expected outputs of wavelet neural network, and W_{ij} and W_{jk} are weights of wavelet neural network. In this paper, Morlet function is selected as the wavelet basis function of wavelet neural network, which is defined as follows:

$$y = \cos(1.75x)\, e^{-x^2/2}. \tag{20}$$

The output layer of wavelet neural network is calculated by (13):

$$y(k) = \sum_{i=1}^{l}\omega_{ik} h(i), \tag{21}$$

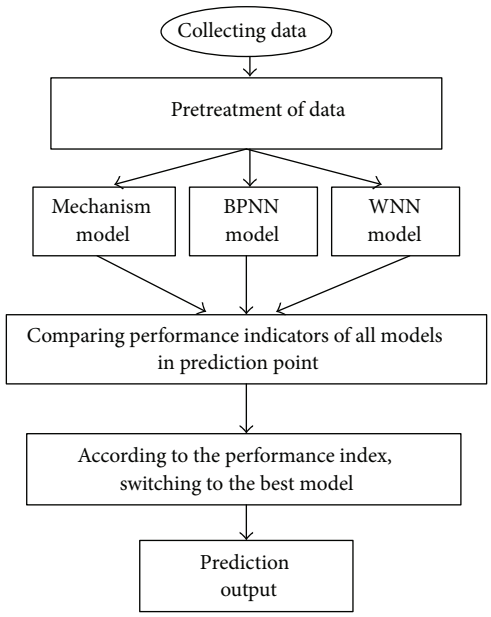

Figure 4: Structure of hybrid multiple soft-sensor model.

where ω_{ik} is the weight form hidden layer to output layer; $h(i)$ is the output of the ith hidden layer nodes; l is the number of hidden layer nodes; m is the number of output layer nodes.

Weights of wavelet basis functions are revised by gradient correction method, which is familiar to BP neural network. With continuous weights correction, the prediction accuracy of wavelet neural network has been improved continuously.

4. Hybrid Multiple Soft-Sensor Model Based on Hysteresis Switching Strategy

4.1. Structure of Hybrid Multiple Soft-Sensor Model. The prediction precision of multiple soft-sensor models is higher than a signal model, but, in each calculation, the multiple models are not suitable for the current actual situation. So using these models to predict the grinding granularity will not only increase the algorithm complexity but also reduce the prediction performance. For this purpose, a multimodel switching thought is proposed, which can dynamically select the proper soft-sensor model. So a hybrid multiple soft-sensor model is set up based on the cuckoo searching algorithm and hysteresis switching strategy, which is made up of mechanism model, BP neural network, and wavelet neural network. Its structure is shown in Figure 4.

4.2. Hysteresis Switching Strategy. Multimodel switching strategy was, at the earliest, used to solve the stability problem of estimation model in adaptive control [13]. Multiple models adaptive control (MMAC) based on index switching strategy was put forward by professor Narendra [14, 15] to ensure that the prediction result is the best prediction of all submodels. At each sampling instant, according to the performance as an indicator, the optimal model is selected as the current model so that the adaptive control of the whole operation is realized. This method has better dynamic performance and faster response speed. Performance indicators are made up of submodel prediction errors, and the current model is a local model that has the minimum performance indicators. The rationality of this method is that the smaller prediction error causes the smaller tracking error [16]. The multimodel switching indicator is represented as follows:

$$J_i(k) = \alpha e_i^2(k) + \beta \sum_{j=1}^{L} w^{k-j} e_i^2(k-j), \quad i = 1, 2, 3, \ldots, N, \tag{22}$$

where $e_i(k) = Y(k) - Y_i(k)$ is the difference between the actual output and the predicted output of the ith model in k instant; α and β are weight coefficients; β determines the proportion of history measurement in performance indicators and represents the effects of the current moment difference and the past moment difference on performance indicators; usually $0.5 < \beta < 1$; N is the number of submodels; L is the error range of the past performance indicators; when the range of the current moment's difference is larger than the current moment's difference, it will have no influence on performance indicators; w represents an error of some past time to now moment; J_i is the switching index representing the divergence between the forecast model and the actual model, so the target of switching strategy is to find a minimum J_i. In the sampling time, according to switching index function, the forecast model is chosen, which is closest to the actual model.

If the difference of k moment and $k-1$ moment is very small, it is meaningless to switch and it will lead to the system being unstable if switching frequently. In order to improve the stability of forecasting system, the switching strategy is replaced by the hysteresis switching strategy; namely, a hysteresis factor is added to performance indicators. For example, the current model is model i; after taking sample of the process output, the switching index of model j is minimal:

$$J_j(k) = \min\{J_i(k)\}, \quad i = 1, 2, \ldots, N. \tag{23}$$

If $j \neq i$, the switching strategy with hysteresis factor ρ ($\rho > 0$) is used to determine whether model i needs to be replaced by model j; if $J_j(k) + \rho \leq J_i(k)$, model i will be replaced by model j, if not, model i will continue to be used. Without frequent switching, the system could keep stable. However the values of α and β are obtained by repeated trial and error in lots of literatures, and then it will reduce the efficiency of the switching and prediction accuracy. In order to save the time of switching and improve the prediction accuracy, the cuckoo search algorithm is used to optimize parameters α and β.

5. Parameters of Hysteresis Switching Strategy Optimized by Cuckoo Search Algorithm

5.1. Cuckoo Search Algorithm. In 2009, the cuckoo searching (CS) algorithm is proposed by Yang of Cambridge University [17, 18]. This algorithm is mainly based on two aspects: cuckoo's parasitic reproduction mechanism and Levy flights

search principle. In nature, cuckoos use a random manner or a quasi-random manner to seek bird's nest location. It is not easy to fall into local optimum compared with other intelligent algorithms and has less parameters. Because it is simple, has less parameters, and is implemented easily, it gradually becomes a new bright spot in the field of swarm intelligence algorithm. Cuckoo search algorithm is inspired by cuckoo parasitic behavior and Levy flights habits. Levy flight is proposed by French mathematician Paul Pierre; without main information or food being randomly distributed, Levy flights model is an ideal searching way for predators. The CS algorithm has been widely used in multiobjective scheduling problem [19], reliability-redundancy allocation problems [20], feed-forward neural network training [21], structural optimization problems [22], fractional delay-IIR filter design [23], global numerical optimization [24], travelling salesman problem [25], satellite image segmentation [26], and so forth.

Many animals and insects' flying behaviors verify the characteristics of Levy flight. In order to simulate cuckoo behaviors, three ideal assumptions are made:

(1) Every cuckoo lays only one egg and randomly places it in a bird's nest.

(2) The cuckoo bird eggs which are placed in the host will hatch and produce the next generation of cuckoo.

(3) The number of nests which the cuckoo can make use of is certain and the probability that cuckoo bird eggs are found is p_a.

On the basis of the above three ideal assumptions, the procedure of CS algorithm is described as follows.

(1) Algorithm Initialization. Suppose $X_0 = (x_1^0, x_2^0, \ldots, x_N^0)$ is N nest positions generated randomly. Then the testing functions are adopted to find the optimal position, and then it will be used in the next generation.

(2) Searching Bird's Nest Position. Through the location updating equation (16), search the nest positions for the next generation of birds. And then the new nest position will be tested by testing function. By comparing this generation testing result with the previous generation testing result, the better result is gotten.

(3) Selecting Bird's Nest Position. $r \in [0, 1]$ is random number. Compare $p_a = 0.25$ with the random number r. If $r > p_a$, the value of x_i^{t+1} is changed randomly; if not, the value of x_i^{t+1} remains unchangeable. Then the changed x_i^{t+1} will be tested by testing function, and the better position pb_t^* is selected by comparing the test result with the previous generation optimal position.

(4) Precision or Iteration Judgment. Calculate $f(pb_t^*)$. If it reaches the target precision or the number of iterations, pb_t^* is the global optimal solution gb; if it is not, pb_t^* will be kept in the next generation and return to Step (2).

It can be seen from the above algorithm steps that the cuckoo search algorithm adopts the Levy flight (global searching) and elite reserving strategy (local searching). Step (3) increases the diversity of solutions and then prevents the algorithm from getting into local optimum. The searching path of cuckoo search algorithm is different from the ordinary algorithms; that is to say that the cuckoo algorithm adopts Levy flight search method, which has strong randomness. Broadly speaking, the step length vector of Levy flight should obey Levy distribution; the migration direction of Levy flight should obey uniform distribution.

Step length vector of cuckoo search algorithm is selected by Mantegna law of Levy distribution characteristics. According to Mantegna law, the size of step length s is defined as follows:

$$s = \frac{u}{|v|^{1/\beta}}, \tag{24}$$

where u and v obey the normal distribution:

$$\begin{aligned} u &\sim N\left(0, \sigma_u^2\right), \\ v &\sim N\left(0, \sigma_v^2\right), \\ \sigma_v &= \left\{ \frac{\Gamma(1+\beta)\sin(\pi\beta/2)}{\Gamma\left[(1+\beta)/2\right]\beta 2^{(\beta-1)/2}} \right\}^{1/\beta}, \\ \sigma_u &= 1. \end{aligned} \tag{25}$$

The searching method of CS algorithm is Levy flight. For example, the ith cuckoo in t generation produces the solution x_i^{t+1} in $t+1$ generation:

$$x_i^{t+1} = x_i^t + \alpha \oplus \text{Levy}(\lambda), \tag{26}$$

where $\oplus$ represents one point to one point multiplication; the step length of Levy(λ) is represented as

$$\text{Levy} \sim u = t^{-\lambda}, \quad (1 < \lambda \leq 3), \tag{27}$$

where α is a control variable of step length vector to control the direction and step size. There is a close relationship between α and the size of searching space. If the searching space is too small and α is too big, some searching space which has optimal solutions will be ignored. The specific relationship between α and the searching space may be described as

$$\alpha = O\left(\frac{L}{10}\right), \tag{28}$$

where L is the size of searching space of the discussed optimization problem.

5.2. Parameters of Hysteresis Switching Strategy Optimized by Cuckoo Search Algorithm. α and β are the components of each dimension of each bird's nest. As a result, they have one-to-one mapping relationship. The fitness function is mean square error of a neural network model. Through optimization of CS, the global optimal value and minimum mean square error are obtained:

$$\text{fitness}(i) = \frac{\sum_{k=1}^{Q}\left(t_k - y_k\right)^2}{Q}, \tag{29}$$

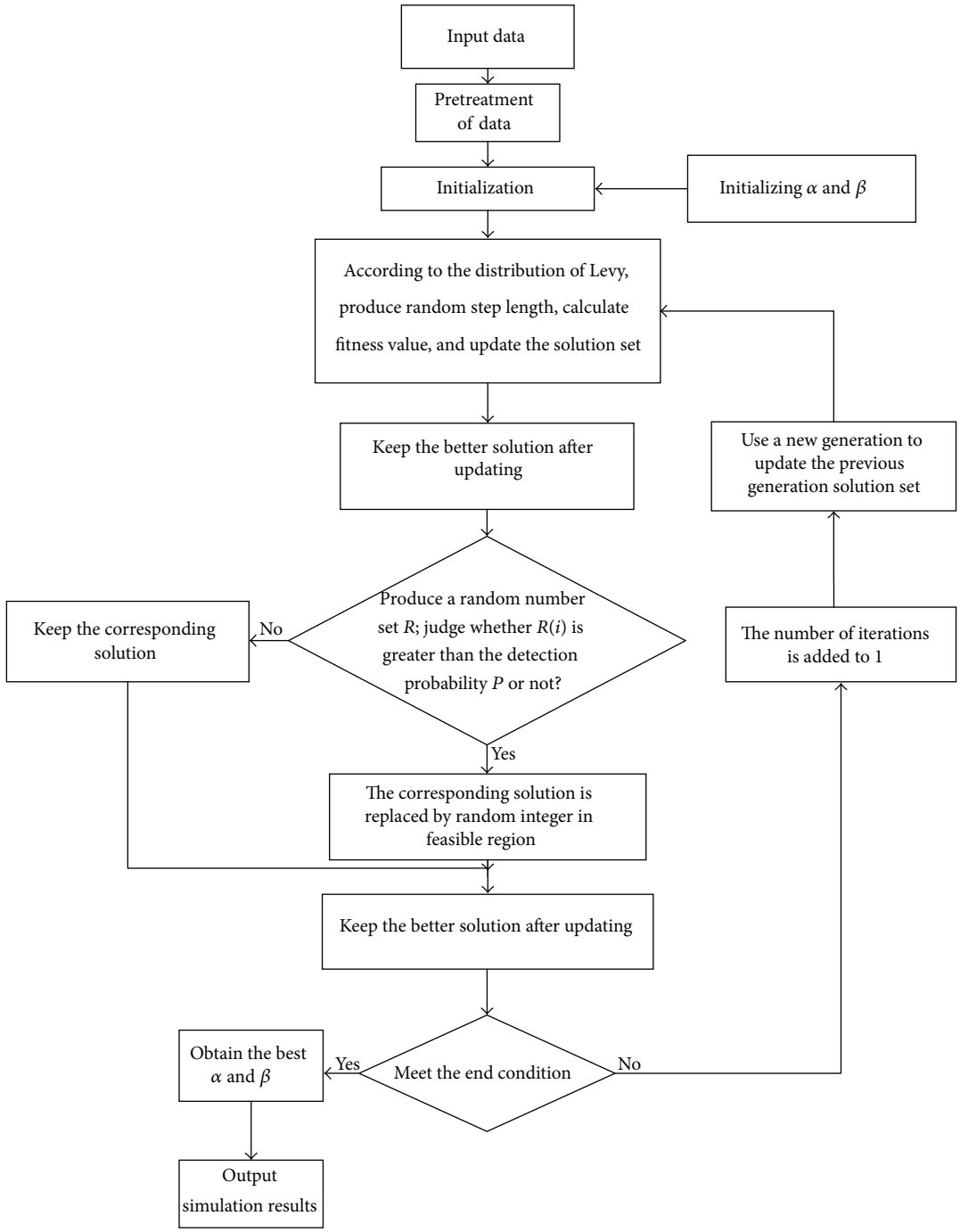

Figure 5: Procedure flowchart of hysteresis switching strategy optimized by CS algorithm.

where i represents the ith bird's nest; Q is the number of training samples; y is the actual output; t is the expecting output.

Before optimizing parameters of hysteresis switching strategy, the parameters of CS need to be determined. The number of iterations $N = 100$; the number of birds' nests $n = 25$; the probability of bird's nest is $p_a = 0.25$; the control variable of step length $T = 0.01$. The procedure flowchart of hysteresis switching strategy optimized by CS algorithm is shown in Figure 5.

6. Simulation

Aiming at the grinding classification process, the grinding granularity soft-sensor model is built. The soft-sensor modeling data are listed in Table 1, where the forehead first 500 groups are training data and the remaining 100 groups are testing data. Before setting up the soft-sensor model, some performance indicators shown in Table 2 are defined to test the performance of soft-sensor models, where $\hat{y}$ is the predictive value and y is the actual value.

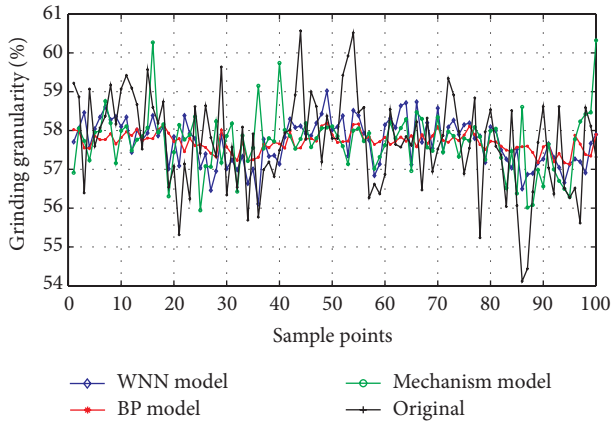

Figure 6: Prediction output curves under three soft-sensor models.

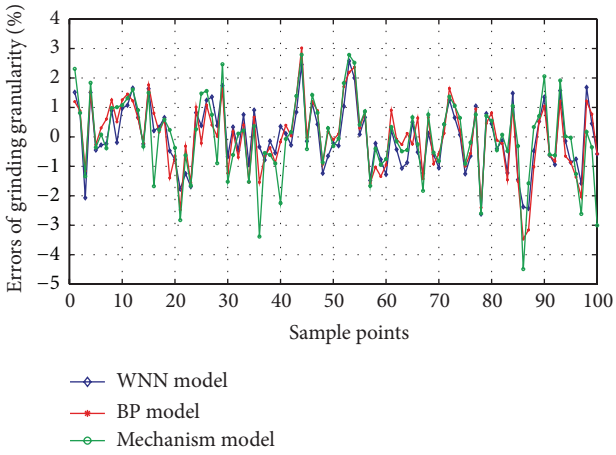

Figure 7: Prediction error curves under three soft-sensor models.

Table 2: Definition of performance indices.

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

To predict the grinding granularity, a mechanism model, BPNN model, WNN model, and hybrid multiple models based on the hysteresis switching strategy are set up. Figure 6 is the prediction curve of three soft-sensor models and Figure 7 is the predictive error under these soft-sensor models. Through the optimization of CS algorithm, the parameters optimum is $\alpha = 0.8$ and $\beta = 0.2$. The hybrid multiple soft-sensor model is set up based on the CS-hysteresis switching strategy, which is then compared with the previous hybrid multiple soft-sensor model based on the pure hysteresis switching strategy. Figures 8 and 9 are simulation comparison results between the hybrid multiple soft-sensor model based on hysteresis switching strategy and

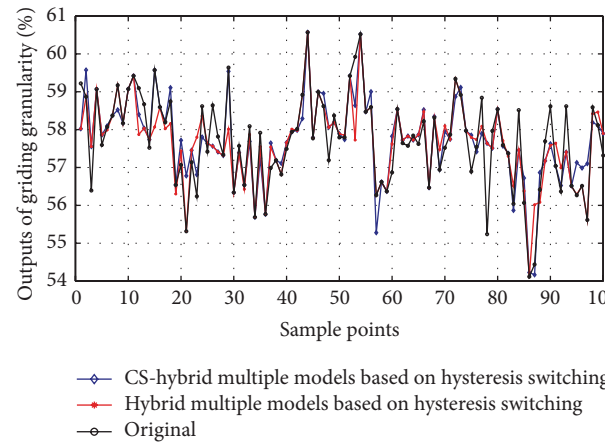

Figure 8: Prediction output curves under hybrid soft-sensor model.

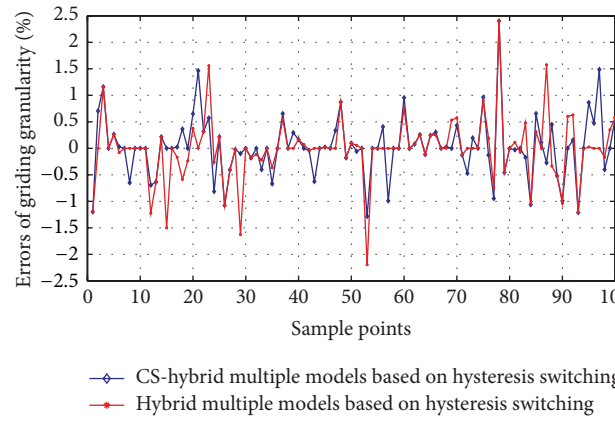

Figure 9: Prediction error curves under hybrid soft-sensor model.

Table 3: Performance comparison of different soft-sensor models.

Soft-sensor model	Performance			
	MPE	MNE	SSE	RMSE
Mechanism model	2.7895	−4.4902	178.002	1.334
BPNN model	3.0189	−3.4657	137.430	1.172
WNN model	2.5782	−2.6213	115.800	1.076
Hybrid model based on HS	2.4001	−2.1960	37.322	0.6109
Hybrid model based on CS and HS	2.4001	1.2835	32.893	0.5735

the hybrid multiple soft-sensor model based on CS-hysteresis switching strategy.

According to the performance indicators defined in Table 2, the performance indicators values of all established soft-sensor models are listed in Table 3. Performance comparisons in the computational time are listed in Table 4. It can be seen from simulation results that the hybrid multiple soft-sensor model based on CS-hysteresis switching strategy is better than other soft-sensor models under four performance indices. The proposed soft-sensor model can realize the prediction of the key technical index and fully meet the control requirements of the grinding process on time.

Table 4: Performance comparisons in computational time for training different predictive models.

Soft-sensor model	BPNN	WNN	Hybrid model based on HS	Hybrid model based on CS and HS
Computational time	13.37	17.84	63.83	127.42

7. Conclusion

For the key technical index (grinding granularity) of the grinding process, a hybrid multiple soft-sensor model based on CS-hysteresis switching strategy is proposed. Through the inferential estimation of the actual operation data, the simulation results show that the hybrid multiple soft-sensor models based on CS-hysteresis switching strategy have good tracking velocity and high prediction accuracy, which can realize the prediction of the key technical index and fully meet the control requirements of the grinding process on time.

Conflict of Interests

The authors declare that they have no conflict of interests.

Authors' Contribution

Jie-Sheng Wang's participated in the concept, design, and interpretation and commented on the paper. A substantial amount of Na-Na Shen's contribution was in the data collection, analysis, algorithm simulation, the draft writing, and critical revision of this paper.

Acknowledgments

This work is partially supported by the National Key Technologies R&D Program of China (Grant no. 2014BAF05B01), the Project by National Natural Science Foundation of China (Grant no. 21576127), 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] J.-S. Wang, X.-W. Gao, and L. Zhang, "Soft-sensor modeling of coal powder granularity of ball mill pulverizing system based on FCM-SVRs method," *Journal of Northeastern University*, vol. 31, no. 5, pp. 613–616, 2010.

[2] X.-D. Zhang and W. Wang, "Beneficiation process of neural networks granularity of soft measurement method," *Control Theory and Application*, vol. 19, no. 1, pp. 85–88, 2002.

[3] J.-J. Ding, H. Yue, Y. Qi, T. Chai, and X. Zheng, "NN soft-sensor for particle size of grinding circuit based GA," *Chinese Journal of Scientific Instrument*, vol. 27, no. 9, pp. 981–984, 2006.

[4] G.-C. He, Y.-P. Mao, and W. Ni, "Grinding size soft sensor model based on neural network," *Metal Mines*, vol. 344, no. 2, pp. 47–49, 2005.

[5] P. Zhou, H. Yue, D.-Y. Zhao, and T.-Y. Chai, "Soft-sensor approach with case-based reasoning and its application in grinding process," *Control and Decision*, vol. 21, no. 6, pp. 646–650, 2006.

[6] H. Chen, L. Ning, and L. Shaoyuan, "Switching multi-model predictive control for hypersonic vehicle," in *Proceedings of the 8th Asian Control Conference (ASCC '11)*, pp. 677–681, IEEE, May 2011.

[7] Y. Li, Y. Fang, J. Li, and S. Shi, "Adaptive backstepping control of hydraulic servo system with input saturation for rolling mill based on multi-model switching," in *Proceedings of the 32nd Chinese Control Conference (CCC '13)*, pp. 3068–3072, IEEE, Xi'an, China, July 2013.

[8] Y.-M. Fang, Z.-Y. Fan, F.-S. Ou, and X.-H. Jiao, "Multi-model switching control with input saturation for hydraulic servo system in rolling mill," *Control Theory & Applications*, vol. 28, no. 3, pp. 438–442, 2011.

[9] J.-S. Wang, N.-N. Shen, and S.-F. Sun, "Integrated modeling and intelligent control methods of grinding process," *Mathematical Problems in Engineering*, vol. 2013, Article ID 456873, 15 pages, 2013.

[10] S. C. Xiu, G. Q. Cai, and C. H. Li, "Study on surface finish mechanism in quick-point grinding," *International Journal of Computer Applications in Technology*, vol. 29, no. 2–4, pp. 163–167, 2007.

[11] H. Ohmori and T. Nakagawa, "Analysis of mirror surface generation of hard and brittle materials by ELID (Electronic In-Process Dressing) grinding with superfine grain metallic bond wheels," *CIRP Annals—Manufacturing Technology*, vol. 44, no. 1, pp. 287–290, 1995.

[12] H. Adeli and X. Jiang, "Dynamic fuzzy wavelet neural network model for structural system identification," *Journal of Structural Engineering*, vol. 132, no. 1, pp. 102–111, 2006.

[13] D. G. Lainiotis, J. G. Deshpande, and T. N. Upadhyay, "Optimal adaptive control: a non-linear separation theorem," *International Journal of Control*, vol. 15, pp. 877–888, 1972.

[14] K. S. Narendra and C. Xiang, "Adaptive control of discrete-time systems using multiple models," *IEEE Transactions on Automatic Control*, vol. 45, no. 9, pp. 1669–1686, 2000.

[15] A. S. Morse, D. Q. Mayne, and G. C. Goodwin, "Applications of hysteresis switching in parameter adaptive control," *IEEE Transactions on Automatic Control*, vol. 37, no. 9, pp. 1343–1354, 1992.

[16] W. Zhao, Y. Niu, and J. Liu, "Adaptive cascade control based multi-model for the superheated steam temperature," *Computer Simulation*, vol. 5, p. 33, 2003.

[17] X.-S. Yang, "Cuckoo search and firefly algorithm: overview and analysis," in *Cuckoo Search and Firefly Algorithm*, vol. 516 of *Studies in Computational Intelligence*, pp. 1–26, Springer, 2014.

[18] X.-S. Yang and S. Deb, "Cuckoo search via Lévy flights," in *Proceedings of the World Congress on Nature and Biologically Inspired Computing (NABIC '09)*, pp. 210–214, IEEE, Coimbatore, India, December 2009.

[19] K. Chandrasekaran and S. P. Simon, "Multi-objective scheduling problem: hybrid approach using fuzzy assisted cuckoo search algorithm," *Swarm and Evolutionary Computation*, vol. 5, pp. 1–16, 2012.

[20] G. Kanagaraj, S. G. Ponnambalam, and N. Jawahar, "A hybrid cuckoo search and genetic algorithm for reliability-redundancy allocation problems," *Computers & Industrial Engineering*, vol. 66, no. 4, pp. 1115–1124, 2013.

[21] E. Valian, S. Mohanna, and S. Tavakoli, "Improved cuckoo search algorithm for feedforward neural network training," *International Journal of Artificial Intelligence & Applications*, vol. 2, no. 3, pp. 36–43, 2011.

[22] A. H. Gandomi, X.-S. Yang, and A. H. Alavi, "Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems," *Engineering with Computers*, vol. 29, no. 1, pp. 17–35, 2013.

[23] M. Kumar and T. K. Rawat, "Optimal design of FIR fractional order differentiator using cuckoo search algorithm," *Expert Systems with Applications*, vol. 42, no. 7, pp. 3433–3449, 2015.

[24] G.-G. Wang, A. H. Gandomi, X. Zhao, and H. C. E. Chu, "Hybridizing harmony search algorithm with cuckoo search for global numerical optimization," *Soft Computing*, pp. 1–13, 2014.

[25] A. Ouaarab, B. Ahiod, and X.-S. Yang, "Discrete cuckoo search algorithm for the travelling salesman problem," *Neural Computing and Applications*, vol. 24, no. 7-8, pp. 1659–1669, 2014.

[26] A. K. Bhandari, V. K. Singh, A. Kumar, and G. K. Singh, "Cuckoo search algorithm and wind driven optimization based study of satellite image segmentation for multilevel thresholding using Kapur's entropy," *Expert Systems with Applications*, vol. 41, no. 7, pp. 3538–3560, 2014.