Flatness Intelligent Control Based on T-S Cloud Inference Neural Network

Xiuling ZHANG,1,2)* Liang ZHAO,1) Jiayin ZANG,1) Hongmin FAN1) and Long CHENG1)

1) Key Laboratory of Industrial Computer Control Engineering of Hebei Province, Yanshan University, Qinhuangdao, 066004 China.
2) National Engineering Research Center for Equipment and Technology of Cold Strip Rolling, Qinhuangdao, 066004 China.

(Received on March 27, 2014; accepted on July 23, 2014)

The accuracy of traditional flatness control methods are limited and it is difficult to establish a precise mathematical model of the rolling mill. In addition, the flatness control system is complex and multivariate. General model approaches can not satisfy the high precision demand of rolling process. In this paper, T-S cloud inference neural network and its stability are proposed. It is constructed by cloud model and T-S fuzzy neural network. The stability of T-S cloud inference neural network is analyzed by Lyapunov method in details. Based on the new network, flatness recognition model and flatness predictive model are established. And they are applied for 900HC reversible cold rolling mill. The flatness control system is designed and a simple controller is developed. Initial parameters of the controller are firstly determined through offline training based on measured data, and then they are optimized online automatically. Genetic Algorithm (GA) is used as the optimizing method which is compared with particle swarm optimization (PSO). The simulation results demonstrate that the flatness control system is effective and has a better precision and robustness.

KEY WORDS: flatness control; T-S cloud inference neural network; cloud model; stability; cold rolling mill; Genetic Algorithm.

1. Introduction

Flatness control is a key technology of the cold rolling strip steel production. It is also a hot research topic in the current international. With the rapid development of economy, the precision of flatness is becoming more and more accurate.1) Strip steel is the main products of rolled steel, and it is widely used in buildings, bridges, ship buildings and cars. Flatness is a significant quality indicator of strip steel. The purpose of flatness research is to solve the accuracy of flatness control. However, the accuracy of traditional flatness control methods is limited, and it is difficult to establish a precise mathematical model of the rolling mill. The rolling process is multivariate, strong-coupling, nonlinear, and time-varying. So, the flatness theory needs to be studied deeply and extensively. Building an accurate, stable, fast model of flatness control has become a development direction of flatness control technology.2)

Artificial intelligence (AI) technique is an efficient alternative tool for complex and nonlinear system modeling in many disciplines.3–7) Specialists and scholars have applied AI to the area of rolling and many new theories and algorithms have been widely used. The development of intelligent control provides new thinking and ways for flatness control technology. Zhang8) utilized internal model control offline. And improved BP algorithm is used for adjusting the weights online. Simulation results showed that the hydraulic bending roll system possessed rapid response and strong robustness and the control effect is ideal. Wang and Li9) combined PSO with GA algorithm and the mixed algorithm was applied for the adaptive PID control of hydraulic bending system. A simulation model was built and a better control performance was got. Based on conventional PID control algorithm, Shan et al.10) built an optimized non-linear combined control rule of P, I, and D by the self-learning ability of neural network and the “conception” abstraction ability of fuzzy control. Simulation results of the model indicated that flatness target value was tracked well with rapid response, small overshoot and strong robust. In order to resolve the problem of low precision, slow speed and bad anti-interference ability in conventional PID flatness control, Jia et al.11) proposed the strategy of single-nerve -cell adaptive PID flatness control based on BP neural network prediction model. Simulation results indicated that the control algorithm can track the target set value of flatness, increase the control precision of the system and accelerate the response speed of the system with strong anti-interference ability. Fan12) designed a fuzzy controller. Through the membership function and quantification factors and scaling factors, the controller was adjusted by the fuzzy control rule table. It has certain adaptive ability.

Based on probability theory and fuzzy mathematics, a new model, called cloud model, is proposed by academician

* Corresponding author: E-mail: zxlysu@ysu.edu.cn
DOI: http://dx.doi.org/10.2355/isijinternational.54.2608
Deyi Li. This model not only reflects the uncertainty of natural language concepts, but also presents the correlation between randomness and fuzziness. It is an uncertainty transformation model between qualitative concept and its quantitative representation which are described by linguistic value. Zhang et al.\textsuperscript{13} proposed a method of implementing intelligent control of inverted pendulum by the use of cloud models. Its control strategy needs no mathematical models of plant but human experience, sensation and logic decision. The experiment results demonstrated that the method had stronger robustness against the variation of states and parameters of plants, and that it had better practicing value and widely prospect. Liu and Li\textsuperscript{14} designed a fuzzy system based on the combination of cloud model theory and neural network. And the system resolved the difficulties of the definition of the subjection function and knowledge regularity in the practical application of fuzzy system. Xu and Fan\textsuperscript{15} proposed an approach of cloud neural network which was used for fault detection and diagnosis of the space propulsion system.

T-S fuzzy neural network was firstly proposed by Takagi and Sugeno. This model utilizes local linear equation to replace constant which is usually in general inference process. So it can generate complex nonlinear function with a few fuzzy rules. When it is used for dealing with multivariable system, the amount of fuzzy rules can be decreased effectively. So, the T-S fuzzy neural network has a giant superiority. Gan et al.\textsuperscript{16} presented a new type of T-S fuzzy neural network. The structure and learning algorithm of the antecedent network is improved and the node number of the fuzzy rule layer is reduced. The shortcoming of redundant fuzzy rule in the T-S fuzzy neural network is overcome. Wang and Zhang\textsuperscript{17} proposed an improved T-S fuzzy neural network with four layers network structure. And the network was applied into the speech identification system.

GA is a stochastic global search and optimization methods developed by imitating mechanism of nature biological evolution. It is an efficient, parallel, global search method. And it can automatically obtain the search process and accumulate knowledge of the search space, and adaptively control the search process so as to achieve the optimal solution. Zhou et al.\textsuperscript{18} proposed an improved genetic algorithm and applied GA to reactive power optimization. The improved crossover operation was in possession of the ability of fast local adjustment, and the improved mutation operation combined sensitivity analysis to generate new individuals. Zhu et al.\textsuperscript{19} researched the improvement of GA and applied the improved algorithm to nonlinear programming problem.

In this paper, T-S cloud inference neural network is proposed. It is a new neural network constructed by cloud model and T-S fuzzy neural network, and the normal cloud is used to take the place of membership function. The rapidity of fuzzy logic and the uncertainty of cloud model for processing data are both taken into account synthetically and the ability of processing data for the new network has been enhanced. At the same time, the structure of T-S cloud inference network is described in detail and Lyapunov method is used to analyze the stability of the network. What’s more, T-S cloud inference neural network is applied for flatness control. The flatness control system is designed and a simple controller is developed. Initial parameters of the controller are firstly determined through offline training based on measured data by 900HC reversible cold rolling mill, and then they are optimized online automatically by GA. The contrast simulation results demonstrate that the flatness control system possessed rapid response and strong robustness. It is an effective system and has a better precision.

The rest of this paper is organized as follows: Section 2 reports the construction of T-S cloud inference neural network. Lyapunov method is utilized to analyze the stability of the network. In section 3, the new network is applied for 900HC reversible cold rolling mill. The flatness control system is designed. Section 4 describes the contrast simulation results of GA and PSO. Section 5 draws conclusions and future works.

2. T-S Cloud Inference Neural Network

2.1. Cloud Model

Concept is the advanced product of intelligence, and it is an object response in the brain. Cloud model is a qualitative and quantitative uncertainty transition model. Quantitative uncertainty is grasped by concept method. And the model realizes uncertainty transition of qualitative concept and quantitative value.

Suppose $U$ is a quantitative domain expressed by precise value, $C$ is a qualitative concept in $U$, if $x \in U$ and $x$ is a random achieve on $C$, and the degree of certainty $\mu(x)$ from $x$ to $C$ is a random number with stable tendency,

$$\mu : U \rightarrow [0,1] \quad \forall x \in U \quad x \rightarrow \mu(x)$$

Then the distribution of $x$ on domain $U$ is called cloud, and each $x$ is called a cloud droplet.

Suppose $U$ is a quantitative domain expressed by precise value, $x \in U$ and $x$ is a random achieve on $C$, $x \sim \mathcal{N}(\bar{x}, \sigma^2)$ and $\sigma^2 = \frac{2}{\pi} \ln(\sigma^2 / \sigma^2)$, the degree of certainty $\mu$ from $x$ to $C$ is

$$\mu = e^{-\frac{(x-\bar{x})^2}{2\sigma^2}} \quad \text{(1)}$$

Then the distribution of $x$ on domain $U$ is called normal cloud.\textsuperscript{20}

The cloud model uses three numerical characteristics to express a concept, they are $\bar{x}$ (Expected value), $\sigma$ (Entropy), $\sigma^2$ (Hyper entropy). $\bar{x}$ is the distributive expectation of cloud droplet in the domain space and the most representative point of the qualitative concept. $\sigma$ is a measurement of the uncertainty of the qualitative concept, and it is decided by randomness and fuzziness of the concept. On the one hand, it is a measurement of randomness of the qualitative concept and reflects the degree of dispersion of cloud droplets; On the other hand, it is a measurement of fuzziness of the qualitative concept and reflects the range of cloud droplets in the domain space. $\sigma$ is a measurement of the uncertainty of $\sigma$.

Normal cloud and three numerical characteristics are shown in Fig. 1.

2.2. The Structure of T-S Cloud Inference Neural Network

T-S fuzzy neural network is constituted by fuzzy logic and neural network. It has the ability of processing the uncertain information, knowledge storage, and self-learning.
The network has been widely used in pattern recognition and intelligent control. Usually, Gauss function is often chosen as membership function:

\[ \mu = e^{-\frac{(x-a)^2}{b^2}} \]  (2)

The shape and distribution of Gauss membership function is shown in Fig. 2.

In this paper, cloud model is utilized to replace the Gauss membership function. The shape and distribution of cloud membership function is shown in Fig. 3. Based on above, T-S cloud inference neural network is constructed. The structure of T-S cloud inference network is shown in Fig. 4. It has two parts: the premise network and the latter network. The premise network is used for matching the premise of cloud fuzzy rules, and the latter network is used for generating the latter of cloud fuzzy rules.

The premise network has three layers:

The first layer is input layer, the input variables are introduced into the premise network through it. It has \( n \) codes.

The second layer is cloud layer. Each node represents a cloud model and the input variables are clouded in this layer. Each input variable is clouded into \( m \) parts. The number of nodes is \( n \times m \). A node in this layer is shown in Fig. 5.

The third layer is cloud inference layer. Each node represents a cloud fuzzy rule. The output of the premise network corresponds to the fitness value of a cloud fuzzy rule. It is usually represented by algebraic product:

\[ \theta_j = u_{1j} \cdot u_{2j} \cdots u_{nj} = \prod_{i=1}^{n} u_{ij}, j = 1,2,\cdots, m \]  (3)

There are \( m \) nodes in this layer.

The latter network has three layers:

The first layer is input layer, and the input variables are transported to the second layer. In this layer, \( x_0 = 1 \) is used to provide the constant term for the latter of cloud fuzzy rule.

The second layer has \( m \) nodes. Each node represents a rule and it is used to calculate the latter of each rule. The outputs:

\[ y^k_j = \sum_{i=0}^{m} x_i \cdot w^k_j \quad (x_0 = 1) \]  (4)

Where \( k = 1,2,\cdots, r; j = 1,2,\cdots, m \).

The third layer is to calculate the total output of the network:

\[ y_k = \frac{\sum_{j=1}^{m} \theta_j \cdot y^k_j}{\sum_{j=1}^{m} \theta_j} \]  (5)

2.3. Optimize the Network by GA

When the \( p \) sample inputs network, the error for each sample is the square sum of each output error:

\[ E_p = \frac{1}{2} \sum_{i=1}^{p} (d_i - y_i)^2 \]  (6)
Where $d_k$ is desired output.

After all the learning sample inputs network, the total error:

$$ E = \sum_{i=1}^{n} E_p = \frac{1}{2} \sum_{i=1}^{n} \sum_{k=1}^{m} (d_k - y_k)^2 \quad \text{(7)} $$

Where $q$ is the count of samples.

GA is an efficient, parallel, global search method. It imitates the biological genetic and evolutionary mechanisms which roots in Darwin’s evolution theory and Mendel’s genetics. The algorithm can automatically obtain the search process and accumulate knowledge of the search space, and adaptively control the search process so as to achieve the optimal solution. In this paper, GA is applied to optimize T-S cloud inference neural network. Equation (7) is selected as fitness function. A mass of samples are used for training the network. When $He$ is determined, the parameters to be optimized in the network are $Ex$, $En$ and the weight $w$ in the latter network.

2.4. Stability Analysis of T-S Cloud Inference Neural Network

In general, a multiple outputs system can be broken up into many single output systems. So, multiple outputs T-S cloud inference neural network can be broken up into many single output T-S cloud inference neural networks. Single output network is analyzed firstly. Then, the method can be popularized for multiple outputs T-S cloud inference neural network.

The discrete cloud fuzzy model of multiple inputs single output T-S cloud inference neural is:

$$ R^i : x_i \in A^i_1 \text{ and } x_{i-1} \in A^i_2 \text{ and } \ldots \text{ and } x_{i-n} \in A^i_n $$

then $x_{i+1} = \omega_1 x_1 + \omega_2 x_{i-1} + \cdots + \omega_n x_{i-n}$

Where: $R^i$ $(i = 1, 2, \ldots, m)$ represents the $i$ th cloud fuzzy rules, and $n$ is the total. $x_i, x_{i-1}, \ldots, x_{i-n}$ are the state variable of system. $A$ is the cloud fuzzy set of state variable. It is the cloud model. $x_{i+1}$ is the output of the second layer in the latter network. $\omega$ is the weight of the latter network. \(^{21}\)

The equation can be written in matrix form:

$$ x(k+1) = W \cdot x(k) \quad \text{(8)} $$

Where: $x(k) \in \mathbb{R}^n$, $W \in \mathbb{R}^n \times \mathbb{R}^n$, $x(k) = [x_1, x_{i-1}, \ldots, x_{i-n+1}]^T$,

$$ W = \begin{pmatrix}
\omega_1^1 & \omega_2^1 & \cdots & \omega_{n-1}^1 & \omega_n^1 \\
1 & 0 & \cdots & 0 & 0 \\
0 & 1 & \cdots & 0 & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & 0 & 1 \\
\end{pmatrix} $$

Total output of the network is:

$$ X(k+1) = \sum_{i=1}^{m} \theta_i \cdot W \cdot x(k) \quad \text{(9)} $$

$\theta_i$ is the fitness of the $i$ th cloud fuzzy rule:

$$ \theta_i = \prod_{j=1}^{n} \mu_j(x_{i-j+1}) $$

So, $\theta_i$ is the function of $x(k)$. Let $\theta_i = g_i(x(k))$, $G[x(k)]$ is:

$$ G[x(k)] = \sum_{i=1}^{m} g_i \cdot W $$

Then, multiple inputs single output T-S cloud inference neural network is:

$$ X(k+1) = G[x(k)] \cdot x(k) \quad \text{(10)} $$

Equation (10) is a state equation of matrix form of nonlinear discrete system, and only the first line of state matrix is effective. Then,

$$ X(k+1) = \sum_{i=1}^{m} \omega_1^i g_i \chi_k + \sum_{i=1}^{m} \omega_2^i g_i \chi_{k-1} + \cdots + \sum_{i=1}^{m} \omega_n^i g_i x_{i-n+1} $$

$$ \text{..........................(11)} $$

Because the expression of each cloud model is not a single one in the whole domain space. It is a piecewise function divided by the value of $x$. So $G[x(k)]$ is a piecewise polynomial matrix. It is hard to analyze the stability directly. For the system, there exists an area $\Omega_2$ around the balance point. In the area $\Omega_2$, each cloud model can be unified to one expression. So, $G[x(k)]$ is a single expression in this area. And it is used to judge the stability of the network. Because of the area $\Omega_2$, $x(k)$ is seen as the constant. So, the whole system is transformed into a linear time-invariant discrete system. \(^{21}\)

$$ X(k+1) = G[x(k)] \cdot x(k) $$

$$ G = \begin{pmatrix}
\sum_{i=1}^{m} \omega_1^1 g_i & \sum_{i=1}^{m} \omega_2^1 g_i & \cdots & \sum_{i=1}^{m} \omega_{n-1}^1 g_i & \sum_{i=1}^{m} \omega_n^1 g_i \\
\sum_{i=1}^{m} g_i & \sum_{i=1}^{m} g_i & \cdots & \sum_{i=1}^{m} g_i & \sum_{i=1}^{m} g_i \\
0 & 1 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 1 \\
\end{pmatrix} $$

The eigenvalues of $G[x(k)]$ in unit circle $\|x\| = 1$ is necessary and sufficient conditions of stability. In the area $\Omega_2$, if every eigenvalue is located in unit circle, then the value range of $x(k)$ will be determined. It is the stability area $\Omega_2$.

When the rank of $G[x(k)]$ is above three. Lyapunov method is utilized to analyze the stability of the network: for any given positive definite symmetric $Q$, there must exist a positive definite symmetric $P$, they satisfy:

$$ G^T P \cdot G = -Q $$
Let:

\[ Q = I = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} \]

If \( P \) is positive, the value range of \( x(k) \) and the stability area \( \Omega_s \) will be obtained.

If above method is used to analyze the stability, an extra condition should be satisfied: when \( x(k) \) is located in area \( \Omega_s \), \( x(k + 1) \) should be located in the same area. Under this condition, \( G(x(k)) \) is a constant matrix. When:

\[
\begin{align*}
|\sum_{i=1}^{m} \omega_i g_i x_i| & \leq \frac{1}{2} \sum_{i=1}^{M} x_{i-1} \cdots x_{i-n} + \frac{1}{2} \sum_{i=1}^{M} x_{i-1} \cdots x_{i-n+1} \\
|\sum_{i=1}^{m} \omega_i g_i x_i| & \leq \frac{1}{2} \left( \sum_{i=1}^{m} \omega_i g_i x_i \right) + \frac{1}{2} \left( \sum_{i=1}^{m} \omega_i g_i x_i \right)
\end{align*}
\]

Where \( M \) is a constant. The task is to find the area \( \Omega_s \). If

\[
|\sum_{i=1}^{m} \omega_i g_i x_i| \leq M (i = 0, 1, 2, \ldots, n-1),
\]

then

\[
|X(k+1)| \leq \left( \sum_{i=1}^{m} \omega_i g_i \right) \left( \frac{1}{2} \left( \sum_{i=1}^{m} \omega_i g_i x_i \right) + \frac{1}{2} \left( \sum_{i=1}^{m} \omega_i g_i x_i \right) \right) M
\]

When

\[
\sum_{i=1}^{m} \omega_i g_i \left( \frac{1}{2} \left( \sum_{i=1}^{m} \omega_i g_i x_i \right) + \frac{1}{2} \left( \sum_{i=1}^{m} \omega_i g_i x_i \right) \right) \leq 1
\]

Apparantly, Eq. (13) is conservative. In a real world application, the value range of \( x(k) \) is obtained flexibly according to Eq. (11). On the basis of Lyapunov system stability definition, the network is asymptotic stability in \( \Omega_s \). So, the stability area is \( \Omega_s \cap \Omega_s \).

To sum up, the stability analysis of multiple inputs single output T-S cloud inference neural network is obtained. The same method can be used to analyze multiple inputs multiple outputs T-S cloud inference neural network.

3. Flatness Control System Design

This paper takes 6-high 900HC reversible cold rolling mill as research object. Based on T-S cloud inference neural network, a closed-loop flatness control system which consists of flatness pattern recognition model, flatness predictive model and flatness controller is designed. The control methods are working roll bending and middle roll shifting. The structure of flatness control system is shown in Fig. 6, and the flow chart of control system is shown in Fig. 7.

3.1. Flatness Pattern Recognition Model

Flatness pattern recognition is a basic part of flatness control. In consideration of the diversity of control method and the requirement of precise control, right-one-third waves and left-one-third waves were introduced on the basis of traditional six basic patterns. So, there are eight common flatness recognition signal basic patterns: left waves, right waves, center waves, double-edge waves, right-one-third waves, left-one-third waves, quarter waves and edge-center waves. The flatness rolled can be formulated as linear combination of basic patterns:

\[
f(y) = u_1 p_1(y) + u_2 p_2(y) + u_3 p_3(y) + u_4 p_4(y) + \ldots
\]

Where:

\[
p_1(y) = 1/2(5y^3 - 3y), p_2(y) = 1/8(15y^4 - 30y^2 + 3)
\]

\( p_i(x), p_2(x), p_3(x), p_4(x) \) represent linear, quadratic, cubic, quartic Legendre polynomial expansion, \( u_1, u_2, u_3, u_4, u_5 \) represent membership degree of flatness deviation. The value reflects the content of each shape deviation, and the symbol reflects the pattern of flatness. When the value is positive, it represents left waves, center waves, right-one-third waves and quarter waves; when the value is negative, it represents right waves, double-edge waves, left-one-third waves and edge-center waves. Suppose the measured flatness stress value to be identified are normalized as: \( Y = \{\sigma(1), \sigma(2), \sigma(m)\} \). The \( k \) th basic pattern is: \( Y_k = [\sigma(1), \sigma(2), \ldots, \sigma(m)] \) \( k = 1, 2, \ldots, 8 \). \( m \) is the amount of measuring section.

The Euclidean distance is regard as the model input in flatness pattern recognition. The Euclidean distance between \( Y \) and \( Y_k \) is:

\[
D_k = \| Y - Y_k \| = \sqrt{\sum_{i=1}^{m} (\sigma(i) - \sigma(i))^2}
\]

According to the requirements of rolling mill implementing agency, left waves and right waves, center waves and double-edge waves, right-one-third waves and left-one-third waves, quarter waves and edge-center waves cannot exist at the same time in the flatness basic pattern identified. If \( D_1 < D_2 \), then \( d_1 = D_1 \), otherwise \( d_1 = -D_1 \); If \( D_3 < D_4 \), then \( d_2 = D_3 \), otherwise \( d_2 = -D_3 \);
If $D_5 < D_6$, then $d_3 = D_5$, otherwise $d_3 = -D_6$.

If $D_7 < D_8$, then $d_4 = D_7$, otherwise $d_4 = -D_8$.

The model of flatness pattern recognition is a four-input and four-output system. The flatness pattern recognition model based on T-S cloud inference neural network is shown in Fig. 8. $d_j (j = 1, 2, 3, 4)$ is the input and $u_j (j = 1, 3, 5, 7)$ is the output.

### 3.2. Rolling Mill Flatness Predictive Model

In order to improve the predictive accuracy, factors of affecting flatness are taken into consideration as many as possible, and the number is selected as seventeen. The basic rolling parameters: initial width of strip $B_0$, entrance average thickness $h_0$, rolling pressure $P$, front tension $T_1$, back tension $T_0$, work roll diameter $D_w$, intermediate roll diameter $D_m$, backup roll diameter $D_b$, rolling temperature $T$, Young’s modulus of elasticity $E$; The flatness adjustment parameters: work roll bending force $F_w$, intermediate roll shifting position $\delta$; The flatness membership degree of $k$ moment: $u_1(k), u_3(k), u_5(k), u_7(k)$. The outputs of predictive model are flatness membership degree of $k + 1$ moment: $u_1(k+1), u_3(k+1), u_5(k+1), u_7(k+1)$. Then, a seventeen inputs four outputs flatness predictive model is constructed by T-S cloud inference neural network. The inputs and outputs are shown as follows: inputs:

$$U = [B, h_0, h_1, P, T_1, T_0, D_w, D_m, D_b, T, E, F_w, \
\delta, u_1(k), u_3(k), u_5(k), u_7(k)]^T$$

outputs:

$$Y = [u_1(k+1), u_3(k+1), u_5(k+1), u_7(k+1)]^T$$

The structure of flatness predictive model is shown in Fig. 9.

### 3.3. Flatness Controller Design

Based on T-S cloud inference neural network, the flatness T-S cloud inference controller is designed. It is a four-input and two-output controller. The inputs are the subtraction of target flatness characteristic value and rolling mill flatness membership degree. The outputs are work roll bending force $F_w$ and intermediate roll shifting position $\delta$. The structure of T-S cloud inference controller is shown in Fig. 10.

Initial parameters of the controller are firstly determined through offline training based on measured data by 900HC reversible cold rolling mill. Then, the controller is applied for flatness control system and it is shown in Fig. 6. GA is utilized to optimize the controller parameters online. According to the structure of T-S cloud inference controller, there are 72 variables need to be optimized. In order to make it more convenient, GA toolbox and real number coding...
method were used. The parameters of GA mainly include population size, stopping generations, crossover probability and mutation probability. These parameters need to be preset: population size is 20; stopping generations are 400; crossover probability is 0.8; mutation probability is 0.2. The optimization objective is to get the minimum of fitness value. Fitness value and current best individual are shown in Figs. 11 and 12.

4. Simulation Results and Discussions

In order to test the effectiveness of the method proposed above, T-S cloud inference neural network is applied for 900HC reversible cold rolling mill. For a kind of steel strip, the rolling specification is: 2.2×662→0.247×662 and the material is SPHC. The rolling basic parameters are shown in Table 1. The experiment samples are all measured continuous data from a steel rolling factory.

4.1. Simulation Test One

Based on T-S cloud inference neural network, flatness predictive model is established. And training samples are normalized. At the exit of first, third and fifth pass, 900HC reversible cold rolling mill is equipped with flatness meter. Then, the predictive flatness is compared with the measured flatness in first, third and fifth pass. The results are shown in Table 2. SSE stands for the square sum of error between predictive outputs and actual outputs. In order to be more lively and clear, flatness three-dimensional diagrams for a
moment are shown in Figs. 13, 14 and 15.

From Table 2 and Figs. 13–15, it can be seen that the predictive flatness has good consistency to the actual flatness. The flatness predictive model established in this paper possesses preferable predictive ability. It provides the basis for flatness control.

4.2. Simulation Test Two

In order to show the control effect of the flatness control system, GA and PSO are utilized to train the controller and the flatness predictive model. The flatness control system is shown in Fig. 6. Under the environment of MATLAB 2010a, the control results of GA and PSO are shown in Table 3. For the sake of direct-viewing and convenience, flatness three-dimensional diagrams for a moment are shown in Figs. 16, 17 and 18.

Table 2 and Figs. 16–18 illustrate that the control effect of GA and PSO. In Table 2, the output flatness membership degrees of GA in fifth pass are smaller than PSO. It indicates that the rolling flatness is more smooth and GA has a better control effect. In Fig. 18, the output flatness can be seen more intuitively. In order to compare the control effect of two algorithms, the flatness curves of the first pass and the fifth pass are drawn in the same figure. It is shown in Fig. 19.

Table 1. The basic rolling parameters.

<table>
<thead>
<tr>
<th>Basic input parameters</th>
<th>Value range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial width of strip B (mm)</td>
<td>460–480</td>
</tr>
<tr>
<td>Entrance average thickness h0 (mm)</td>
<td>0.24–4.0</td>
</tr>
<tr>
<td>Exit average thickness h1 (mm)</td>
<td>0.24–1.5</td>
</tr>
<tr>
<td>Rolling pressure P (MPa)</td>
<td>0–13</td>
</tr>
<tr>
<td>Front tension T1 (N)</td>
<td>4 000–80 000</td>
</tr>
<tr>
<td>Back tension T0 (N)</td>
<td>4 000–80 000</td>
</tr>
<tr>
<td>Work roll diameter Dw (mm)</td>
<td>245–270</td>
</tr>
<tr>
<td>Intermediate roll diameter Dm (mm)</td>
<td>320–340</td>
</tr>
<tr>
<td>Backup roll diameter Db (mm)</td>
<td>790–850</td>
</tr>
<tr>
<td>Rolling temperature T (°C)</td>
<td>55–56</td>
</tr>
<tr>
<td>Young’s modulus of elasticity E (MPa)</td>
<td>21 000</td>
</tr>
<tr>
<td>Work roll bending force Fw (T)</td>
<td>–25.969–40.8</td>
</tr>
<tr>
<td>Intermediate roll shifting position δ (mm)</td>
<td>–200–200</td>
</tr>
</tbody>
</table>

Table 2. The results of predictive flatness and actual flatness.

<table>
<thead>
<tr>
<th>Pass</th>
<th>Predictive outputs</th>
<th>Actual outputs</th>
<th>SSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>First pass</td>
<td>$u_1 = 0.6404$</td>
<td>$u_1 = 0.6415$</td>
<td>0.0016</td>
</tr>
<tr>
<td></td>
<td>$u_3 = –0.162$</td>
<td>$u_3 = –0.1489$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$u_5 = –0.5683$</td>
<td>$u_5 = –0.6084$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$u_7 = –0.3132$</td>
<td>$u_7 = –0.3012$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$u_1 = 0.8716$</td>
<td>$u_1 = 0.87$</td>
<td></td>
</tr>
<tr>
<td>Third pass</td>
<td>$u_3 = –0.2159$</td>
<td>$u_3 = –0.212$</td>
<td>0.000102</td>
</tr>
<tr>
<td></td>
<td>$u_5 = –0.786$</td>
<td>$u_5 = –0.7809$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$u_7 = –0.168$</td>
<td>$u_7 = –0.1604$</td>
<td></td>
</tr>
<tr>
<td>Fifth pass</td>
<td>$u_1 = 0.6336$</td>
<td>$u_1 = 0.6503$</td>
<td>0.0022</td>
</tr>
<tr>
<td></td>
<td>$u_3 = –0.0751$</td>
<td>$u_3 = –0.1155$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$u_5 = –0.4773$</td>
<td>$u_5 = –0.4736$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$u_7 = –0.2812$</td>
<td>$u_7 = –0.2638$</td>
<td></td>
</tr>
</tbody>
</table>

Fig. 13. Flatness three-dimensional diagrams for a moment in first pass: (a) two-dimensional diagram of predictive flatness and actual flatness; (b) three-dimensional diagram of predictive flatness; (c) three-dimensional diagram of actual flatness.

Fig. 14. Flatness three-dimensional diagrams for a moment in third pass: (a) two-dimensional diagram of predictive flatness and actual flatness; (b) three-dimensional diagram of predictive flatness; (c) three-dimensional diagram of actual flatness.
Fig. 15. Flatness three-dimensional diagrams for a moment in fifth pass: (a) two-dimensional diagram of predictive flatness and actual flatness; (b) three-dimensional diagram of predictive flatness; (c) three-dimensional diagram of actual flatness.

Fig. 16. Flatness three-dimensional diagrams for a moment in first pass: (a) two-dimensional diagram of GA and PSO control effects; (b) three-dimensional diagram of GA control effect; (c) three-dimensional diagram of PSO control effect.

Fig. 17. Flatness three-dimensional diagrams for a moment in third pass: (a) two-dimensional diagram of GA and PSO control effects; (b) three-dimensional diagram of GA control effect; (c) three-dimensional diagram of PSO control effect.

Fig. 18. Flatness three-dimensional diagrams for a moment in the fifth pass: (a) two-dimensional diagram of GA and PSO control effects; (b) three-dimensional diagram of GA control effect; (c) three-dimensional diagram of PSO control effect.
Table 3. The control results of GA and PSO.

<table>
<thead>
<tr>
<th>Pass</th>
<th>GA</th>
<th>PSO</th>
</tr>
</thead>
<tbody>
<tr>
<td>u₁ = -0.486</td>
<td>u₁ = -0.7022</td>
<td></td>
</tr>
<tr>
<td>u₁ = -0.4776</td>
<td>u₁ = -0.2315</td>
<td></td>
</tr>
<tr>
<td>u₁ = -0.1141</td>
<td>u₁ = -0.3541</td>
<td></td>
</tr>
<tr>
<td>u₁ = -0.5111</td>
<td>u₁ = -0.6906</td>
<td></td>
</tr>
<tr>
<td>u₁ = -0.2928</td>
<td>u₁ = -0.0103</td>
<td></td>
</tr>
<tr>
<td>u₁ = 0.0945</td>
<td>u₁ = 0.0657</td>
<td></td>
</tr>
<tr>
<td>u₁ = -0.2153</td>
<td>u₁ = -0.0064</td>
<td></td>
</tr>
<tr>
<td>u₁ = 0.1854</td>
<td>u₁ = 0.5869</td>
<td></td>
</tr>
<tr>
<td>u₁ = -0.1051</td>
<td>u₁ = 0.4714</td>
<td></td>
</tr>
<tr>
<td>u₁ = 0.1205</td>
<td>u₁ = 0.1634</td>
<td></td>
</tr>
<tr>
<td>u₁ = -0.2997</td>
<td>u₁ = -0.3394</td>
<td></td>
</tr>
</tbody>
</table>

Figure 19 illustrates the flatness comparison curves of the first pass and the fifth pass. In the same circumstances, the fifth pass flatness deviation compared with the first pass decreases 85.85 percent by T-S cloud inference controller based on GA. While, flatness deviation decreases 39.79 percent by T-S cloud inference controller based on PSO. It can be seen that T-S cloud inference controller based on GA is more superior. So, the flatness control system designed by T-S cloud inference neural network based on GA is an effective flatness control method and has a better precision. The training process is fast and it has a strong robustness.

5. Conclusions and Future Research

This paper proposes a new neural network, called T-S cloud inference neural network. It is a multiple inputs multiple outputs network. The cloud model is combined with T-S fuzzy neural network. The rapidity of fuzzy logic and the uncertainty of cloud model for processing data are both taken into account synthetically. The stability of the network is analyzed by Lyapunov method. And then, flatness pattern recognition model and flatness predictive model are established. By using T-S cloud inference controller, a flatness control system is constructed completely. The system is applied for 900HC reversible cold rolling mill. Simulation results demonstrate that GA is a better algorithm to optimize the T-S cloud inference neural network compared with PSO. It can make the flatness control system reach higher precision and better control effect. The flatness control system based on T-S cloud inference neural network optimized by GA is a new flatness control method. The future work will concentrate on flatness control system of the visual. The flatness control interface is utilized to simulate the whole rolling process. And the control results will be showed lively and convenient with pressing a button.

Acknowledgments

The authors wish to thank the anonymous reviewers for their constructive comments and suggestions. Those comments are all valuable and very helpful for revising and improving our paper, as well as the important guiding significance to our research. This work is supported by the National Natural Science Foundation of China (Grants No. 61074099) and Cultivation Program Project for Leading Talent of Innovation Team in Colleges and Universities of Hebei Province (No. LJRC013. Automation technology and application of continuous casting and steel rolling).

REFERENCES

24) X. Han, H. Liu and C. Jia: Iron Steel, 45 (2010), 56.