Neuro-Fuzzy Tension Controller for Tandem Rolling

F. Janabi-Sharifi, *Senior Member, IEEE*
Department of Mechanical, Aerospace, and Industrial Engineering, Ryerson University
Toronto, Ontario, Canada, M5B 2K3
Email: fsharifi@acs.ryerson.ca

J. Liu
Department of Electrical and Computer Engineering, University of Waterloo
Waterloo, Ontario, Canada N2L 3G1
Email: j5liu@engmail.uwaterloo.ca

Abstract-- **A fuzzy logic controller (FLC) is designed to maintain constant tension for tandem rolling mills. Envisioning fuzzy inference system as neural network and introducing tutor, backward propagation algorithm is used as self-organization technique for FLC to approach the best parameters under supervision. Simulation results exhibit the generalization and adaptivity of neuro-fuzzy controller in offline tuning.**

Keywords-- **tension control, fuzzy logic, neuro-fuzzy system, rolling mill.**

I. INTRODUCTION

Tandem rolling mills usually consist of a number of mill stands arranged in alignment. Various cross-sectional long metal workpieces are reduced step-by-step under high pressure as they proceed through mill stands sequentially [1]. To meet the dimension requirement and regulate mass flow, automatic gage controllers (AGCs) and automatic speed reguators (ASRs) are employed respectively (Fig. 1).

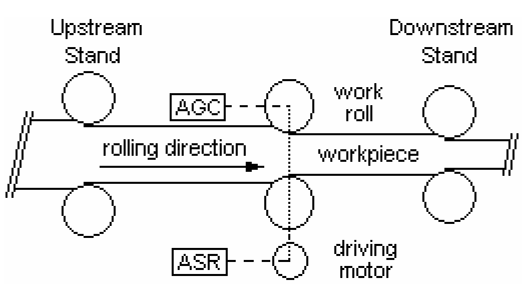

Fig. 1 A rolling mill stand

A specific problem associated with tandem rolling mills is the presence of tension, a longitudinal force inside the workpiece results from the unequal mass flow of two adjacent mill stands. To optimize the performance of AGC and ASR, it is desirable to keep tension constant via addtitional control action. However, resonance effect, *i.e.*, activities of AGC and ASR will incur the variation of tension and in turn tension adjustment will worsen gage and speed control, will perplex the situation especially in harsh noisy environments.

II. FUZZY TENSION CONTROLLER

Conventional tension control schemes need exact mathematical models and complete knowledge of real-time operation [5]. However, it is difficult to identify rolling processes from the measurement of rolling data because of the complicated characteristics of rolling stands, resonance effect, noisy environment and strong demand for instrumentation. Multivariable controllers, based on advanced control theory, have been proposed. However, they are difficult to be implemented and configured [2].

In practice, human experts (operators) can manipulate the rolling stands and manage the inter-stand tension in almost satisfactory manner by manual intervention. This fact motivated us to formalize expert knowledge as fuzzy IF-THEN rules [3] to develop a controller based on fuzzy logic to circumvent incomplete plant knowledge and inexact operation information. This controller will emulate the way that human brain processes ambiguous information with intuition and experience. Greatly inspired by these merits and the competence of human operators, an attempt is made to apply fuzzy logic to tension control for complex nonlinear rolling process.

To eliminate the tension variation, a fuzzy logic controller (FLC) is superimposed on the ASR to adjust the reference speed of ASR. In this scheme, armature current of roll stand driving motor is used as a rough direct indicator of tension between controlled and downstream stands. Therefore terms tension and current will be used interchangeably. The normal value of target tension i_t is sampled before the workpiece threading stand enters downstream one. Comparing this value with the feedback of actual tension i_a, provides error for FLC to remove tension fluctuation. The input signals of FLC are input1 $x_1 = i_e = i_t - i_a$, *i.e.*, current error, and input2 $x_2 = \Delta i_e$, *i.e.*, current error variance. The output signal of FLC is output $y = \Delta v_f$, *i.e.*, speed reference variance.

The design of a FLC necessitates the selection of such control elements and parameters as scaling factors (SFs) for input/output signals, rule base, (de)fuzzification methods and operations for fuzzy reasoning, which include implication, compositional and aggregation operations [4].

III. TRAINING

The performance of FLC heavily relies on the configuration of the factors mentioned. The options of control elements of FLC are numerous and the selection is demanding and plant related. Another difficulty arises in human matter: the control pattern of operator resists systematic articulation, which is necessary in FLC reasoning.

The learning capability of human brain nerve system evokes the effort to introduce artificial neural network for generalization and adaptation of FLC under a tutor's guidance. To endow the supervised learning capability of neural network, backward propagation algorithm (BPA) is employed. The neuro-fuzzy controller (NFC) will tune the parameters of FLC based on the principle of function approximation [3 and 6]. The desired response curve is generated by a tutor and will be used for guiding the learning/tuning process.

Based on these configurations and principles, a neuro-fuzzy tension control system is constructed as illustrated in Fig. 2. In this work, off-line tuning performance of NFC will be studied and its effectiveness will be verified through the step-responses as step inputs are typical inputs for signal corrections in rolling mills.

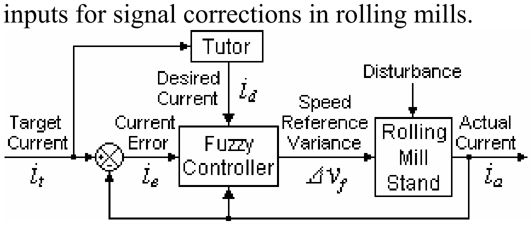

Fig. 2 Neuro-Fuzzy Tension Control System

The ideal tutor is target tension reference. However, in most cases like step input signal, the initial ideal tracking speed is infinite, which is not realistic. In this study, a dynamic linear system is utilized to generate desired tension control action. This kind of tutor would facilitate the training of FLC since the specifications of response can be expressed by standard formulae easily. Furthermore, the tutor shares the same input signal as fuzzy controller. In this way, the tutor can be synchronized with the controller. If the target current is changed, the instruction from tutor will be modified accordingly.

IV. NEURO-FUZZY TENSION CONTROLLER

A. Controller Configuration

The humanoid reasoning in fuzzy logic controller can be observed from reflecting upon expert knowledge and engineering experience, articulated in the design course.

The use of FLC in looper tension control for rolling mills has been recently proposed [5]. Based on this work, the FLC is initially configured following these two guidelines:

(1) Constraint

In light of the derivation requirement, the zero-order Sugeno-type fuzzy inference system is used, *i.e.*, input linguistic variables are fuzzified as fuzzy variables while output linguistic variable takes fuzzy singleton as value. In turn, the defuzzification method is weighted average, and product is used as the aggregation operations of antecedents.

(2) Simplicity

For computation and expression simplicity, min is selected for implication operation, max-min for compositional operation, and union for aggregation operation of consequents. The measurement of input signal is interpreted as fuzzy singleton by ritual.

In this study, the piecewise continuous MF's, in the form of trapezoids and triangles, are used and defined by points (a=left bottom, b=vertex, c= right bottom) and (a=left bottom, b=left top, c= right top, d=right bottom) respectively (Table 1).

Table 1 Configuration of MFs and SFs.

	Input1	**Input2**	**Output**
SF	1	3	0.1
NB	(-1, -1, -0.6, –0.3)	-	-0.75
NM	-	-	-0.5
NS	(-0.6, -0.3, 0)	-	-0.25
N	-	(-1, -1, -0.5, 0)	-
Z	(-0.3, 0, 0.3)	(-0.5, 0, 0.5)	0
P	-	(0, 0.5, 1, 1)	-
PS	(0, 0.3, 0.6)	-	0.25
PM	-	-	0.5
PB	(0.3, 0.6, 1, 1)	-	0.75

Applying plant information and human control expertise, a set of intuitive fuzzy rule base is built up as shown in Table 2.

Table 2 Rule Base.

Ouptupt		Input1				
		NB	**NS**	**Z**	**PS**	**PB**
Input2	**N**	PB	PM	PS	NS	NB
	Z	PB	PM	Z	NM	NB
	P	PB	PS	NS	NM	NB

The control surface in Fig. 3 shows the performance of FLC with these configurations.

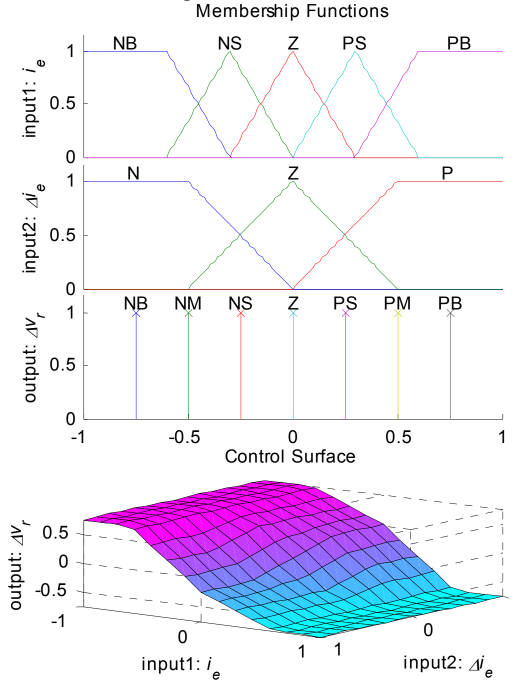

Fig. 3 MF and Control Surface of FLC.

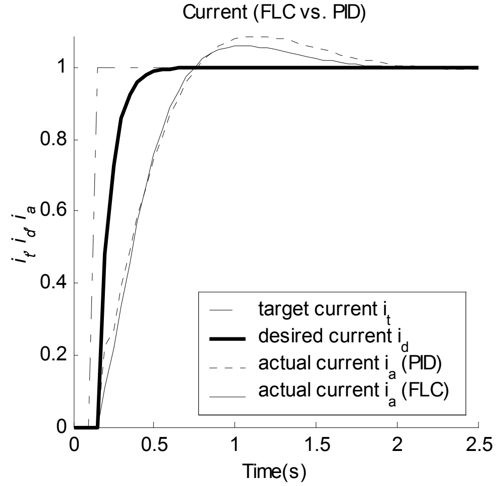

Fig. 4 Unit Step Response of FLC

As an assessment of control activity, a PID controller based on the same tension regulation scheme is also constructed with P = 0.0008, I = 0.1 and D = 0.0001. The comparability of these two controllers can be verified in the unit step response of actual current (Fig. 4).

B. *Neural Network Structure*

When the fuzzy inference system is envisioned as a set of interconnected processing elements and the parameters are visualized as strength of connection, then the fuzzy inference system can be conceived as a feed forward neural network. The tunable elements include the SF's and MF's parameters for both input and output signals of FLC, and rule weights.

With the configuration of FLC depicted in 4.1, the artificial edition of neural network can be constructed as Fig. 5. The difference of the shown network with the conventional artificial neural network is in its architecture, *i.e.*, the node and layer numbers, are determined by fuzzy inference system, and the connections among nodes, *i.e.*, the signal weighting, collecting and processing, are fuzzy operations.

Limited by the availability of probability distribution characteristics and in turn the expectation of data, performance index [6] is approximated by:

$$P(x_1, x_2) = e^2(t) = (i_d(t) - i_a(t))^2$$

With every pair of actual/desired current $[i_a(t), i_d(t)]$ as training data, FLC parameters are adjusted in the reverse layer to reduce the performance index.

C. *Parameter Protection*

Since the amount of training data is finite, two important issues in training of NFC must be addressed: generalization and overfit. Especially, overfit can cause undesirable control activity and would result in oscillation and steady-state error. However, the structure of network is stiff, *i.e.*, the node and layer numbers are determined by fuzzy inference system. Preventive measures should be taken in case of the occurrence of unreasonable parameter tuning. In this work, these two problems are solved from response specification perspective: the protections are provided for FLC parameters during tuning to ensure the stability and final steady-state precision. For triangle/trapezoid MF, the parameters c of NS, d of N, b of Z, a of P and a of PS regulate the steady-state response. If the initial values are chosen as 0's, then these parameters are not to be tuned.

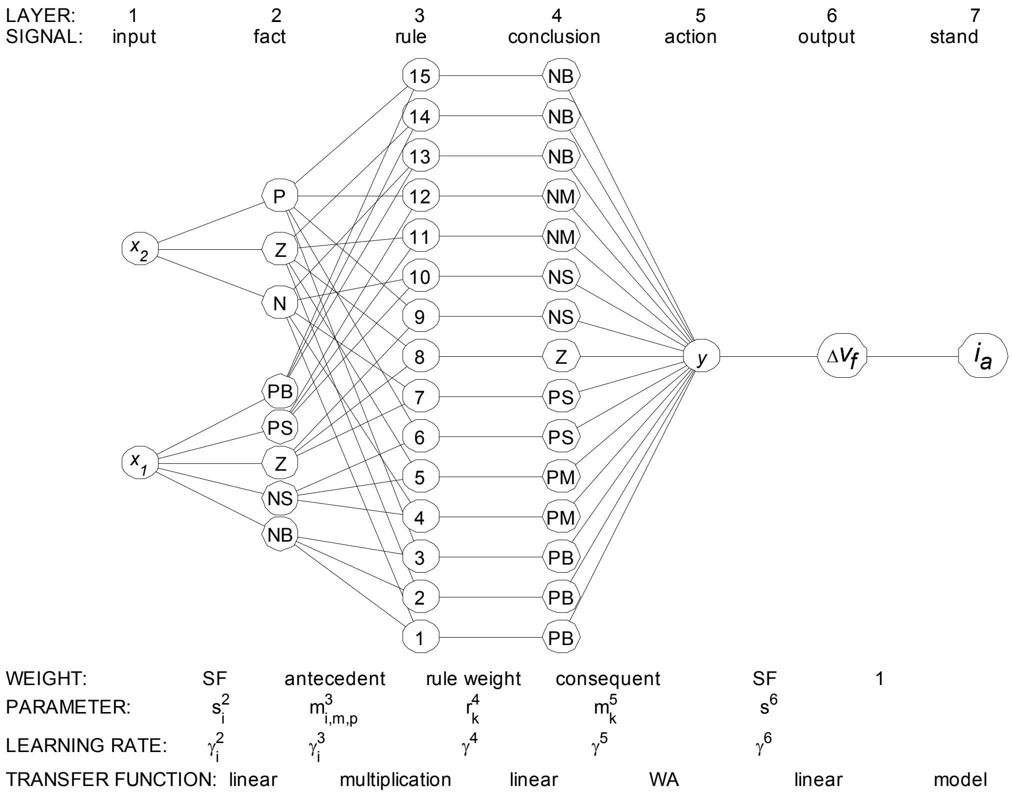

Fig. 5 Neuro-fuzzy controller structure.

D. Forward-Backward Propagation Algorithms

The data in neural network flows in two directions: the control signal flows forward from layer 1 to layer 7 and the error signal flows backward from layer 7 to layer 1.

In neuron-based fuzzy inference system, the reasoning procedure can be articulated by forward propagation algorithm as shown in Table 3. Meanwhile, the tutor makes use of the same target current and gives the desired current.

For every node, the error signal flows from the last layer can be elicited and the parameters are updated as depicted in Table 4.

V. SIMULATION RESULTS

In this study, the simulation of NFC is conducted on Matlab 6. To delve the efficiency of BPA before practical application, a discrete linear dynamic system in the form of:

$$\frac{y(z)}{u(z)} = \frac{-28.6666\mathrm{z}^{-1} + 23.9194\mathrm{z}^{-2}}{1\text{-}0.87224\mathrm{z}^{-1} + 0.0027044\mathrm{z}^{-2}\text{-}0.039773\mathrm{z}^{-3}}$$ is

used as a block box model of rolling mill stand model, which is identified by means of generalized least-squares method using data collected from field (Sidebeck, plant, Quebec). The sampling of target current is simulated by a signal generator.

In the following off-line tuning, as tutor, a first-order linear dynamic system generates target curve with 0.3s settling time and 0 overshoot. For every epoch, the system along with tutor runs for 2.5s. Apart from parameter protection, another way to give neural network self-limit ability is choosing learning rates in conjunction with setting termination condition like error tolerance and epochs in case of overfit. After trial-and-error test, the error tolerance is set as 0.001. The learning rates are chosen as follows: γ_1^2=3, γ_2^2=30, γ_1^3=0.1 γ_2^3=1, γ^4=0, γ^5=0.01, and γ^6=0.3. The tuning of rule weights is turned off and the nodes in rule layer are treated as dead neurons because their validity is questionable.

With these learning parameters, the tuning process indicated by mean square error (MSE) for 45 epochs is shown in Fig. 6.

It can be seen that the BPA steers the error all the way down under the tutor's guide; when the tuning stops at epoch 45, the final MSE reduces from 0.025 to 0.0018.

The parameters of FLC after tuning are listed in Table 5. As shown in Fig. 7, the performance of tension control is improved dramatically after tuning.

Table 3 Forward Propagation Algorithm.

Layer	Formulae
1	$x_1 = i_e(t)$ and $x_2 = \Delta i_e(t)$ $y_k^1(t) = x_k(t)$ k = 1, 2.
2	$x_k^2(t) = s_k^2(t) \times y_k^1(t)$ $y_k^2(t) = x_k^2(t)$ k = 1, ..., 8. $s_k^2(t)$ is the SF for input as weight
3	$x_k^3(t) = \prod_{i=1}^{2} \mu_{i,j}^3(y_i^2(t), m_{i,j}^3(t))$ $y_k^3(t) = x_k^3(t)$ k = 1, ..., 15. $\mu_{i,j}^3(y_i^2(t), m_{i,j}^3(t))$ is the degree of $y_i^2(t)$ in j[th] fuzzy variable, which is designated by k[th] rule, with MF parameters $m_{i,j}^3(t)$ for input i.
4	$x_k^4(t) = r_k^4(t) \times y_k^3(t)$ $y_k^4(t) = x_k^3(t)$ k = 1, ..., 15. r_k^4 is rule weight.
5	$x^5(t) = \sum_{j=1}^{15} m_i^5(t) \times y_j^4(t)$ $y^5(t) = \dfrac{x^5(t)}{\sum_{j=1}^{15} y_j^4(t)}$ m_j^5 is j[th] fuzzy singleton, which is designated by j[th] rule.
6	$x^6(t) = s^6(t) \times y^5(t)$ $y^6(t) = x^6(t)$ $s^6(t)$ is the SF for output as weight.
7	$y^7(t) = i_a(t)$
Where: $x_n^l(t)$ is the net input of node *n* in layer l. $y_n^l(t)$ is the output of node n in layer l.	

Table 4 Backward Propagation Algorithm.

Layer	Formulae
7	$J^7(t) = -2 \times e(t) \times \dfrac{\partial i_a(t)}{\partial \Delta v_r(t)}$
6	$J^6(t) = J^7(t)$ $s^6(t+1) = s^6(t) - \gamma^6 \times J^6(t) \times y^5(t)$
5	$J^5(t) = \dfrac{1}{\sum_{j=1}^{15} y_j^4(t)} \times s^6(t) \times J^6(t)$ $m_k^5(t+1) = m_k^5(t) - \gamma^5 \times J^5(t) \times \overline{y_i^4(t))}$ k = 1, ..., 7.
4	$J_k^4(t) = m_i^5(t) \times J^5(t)$ $r_k^4(t+1) = r_k^4(t) - \gamma^4 \times J_k^4(t) \times y_k^3(t)$ k = 1, ..., 15.
3	$J_r^3(t) = r_r^4(t) \times J_r^4(t)$ $m_{i,m,p}^3(t+1) = m_{i,m,p}^3(t) - \gamma_i^3$ $\times \dfrac{\partial \mu_{i,m}^3(y_i^2(t), m_{i,m}^3(t))}{\partial m_{i,m,p}^3(t)}$ $\overline{\times J_r^3(t) \times \mu_{3-i,j}^3(y_{3-i}^2(t), m_{3-i,j}^3(t))}$ r = 1, ..., 15. i = 1, 2. m = 1, ..., 5 (input1) or 3 (input2). p = 1, ..., 3 (triangle) or 4 (trapezoid).
2	$J_{i,k}^2(t) = \sum_j \Big(\dfrac{\partial \mu_{i,k}^3(y_i^2(t), m_{i,k}^3(t))}{\partial y_i^2(t)}$ $\times \mu_{3-i,j}^3(y_{3-i}^2(t), m_{3-i,j}^3(t)) \times J_r^3(t)\Big)$ $s_i^2(t+1) = s_i^2(t) - \gamma_i^2 \times \overline{J_{i,k}^2(t) \times y_i^1(t))}$ i = 1, 2. k = 1, ..., 5 (input1) or 3 (input2)
Where: $J_n^l(t) = \dfrac{\partial P(x_1, x_2)}{\partial x_n^l(t)}$ is the sensitivity of node *n* in layer l. $\overline{}$ is average operation.	

Table 5 Parameters of MF's after Tuning.

	Input1	Input2	Output
SF	1.336	2.381	0.2502
NB	(-1, -1, -0.6, –0.3)	-	-0.7523
NM	-	-	-0.5017
NS	(-0.6, -0.2859, 0)	-	-0.2549
N	-	(-1, -1, -0.4380, 0)	-
Z	(-0.3, 0, 0.2919)	(-0.6082, 0, 0.5012)	0
P	-	(0, 0.5034, 1, 1)	-
PS	(0, 0.2957, 0.6066)	-	0.2498
PM	-	-	0.5025
PB	(0.2748, 0.5822, 1, 1)	-	0.75

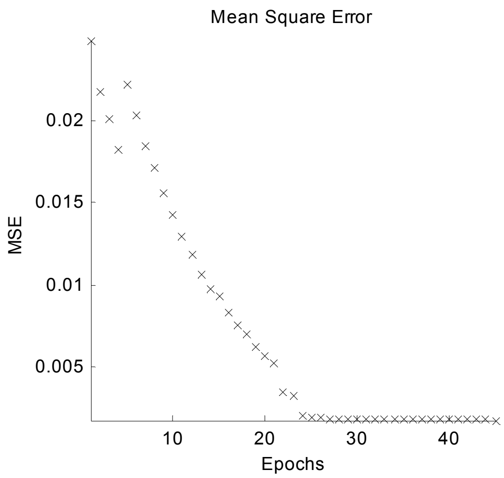

Fig. 6 Tuning performance

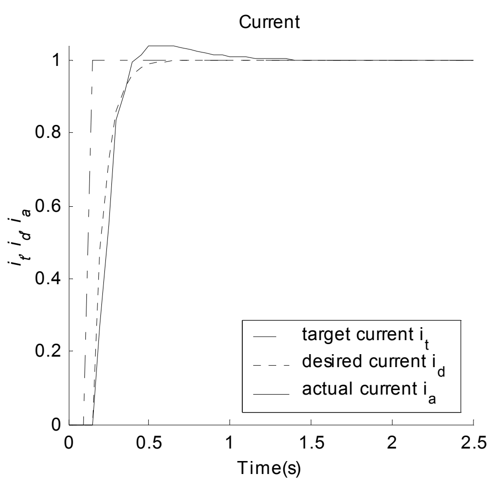

Fig. 7 Response after Offline Tuning

VI. CONCLUSIONS

In this paper, the application of fuzzy-reasoning-based neural net for the tension control of rolling mill was investigated. Aiming at dealing with unstructured FLC design, incomplete plant knowledge and uncertain environment, best parameters around initial values are elicited via BPA. Preliminary tuning simulations on a single stand indicate the successful self-learning of NFC.

BPA necessitates the subtle and strenuous selection of learning parameters. In parameter selections a trade-off between learning speed and performance must be made. Apart from local optima, a major disadvantage of NFC is the derivative computation and continuity of control space that BPA calls for. These constraints limit the space for options of fuzzy control elements.

The supervision for NFC marks the boundaries of effective tuning. One of the most beneficial consequences of introducing tutor into tuning is that the amount of data for training is determined by running time and only limited by sampling time. The instruction from tutor contains valuable performance information over whole response period with high resolution.

ACKNOWLEDGMENT

This work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) through Collaborative Research and Development Grant CRDPJ 234028-99, and by Materials and Manufacturing Ontario Collaborative Research Grant IC403. The authors would like also to thank the assistance of Quad Engineering Inc. in providing data and for the useful comments.

REFERENCES

[1] V. B. Ginzburg, *Steel-rolling Technology: Theory and Practice*, New York, M. Dekker, 1989.
[2] V. B. Ginzburg, *High-Quality Steel Rolling, Theory and Practice,* New York, Marcel Dekker, 1993.
[3] C. T. Lin and C. S. George Lee, *Neural Fuzzy Systems, A Neuro-Fuzzy Synergism to Intelligent Systems,* Upper Saddle River: Prentice Hall, Inc., 1996.
[4] L. Reznik, "Fuzzy controller design: Recommendations to the user," *Second Int. Conf. on Knowledge-Based Intelligent Electronic Systems,* Adelaide, Australia, pp. 609-616, 1998.
[5] F. Janabi-Sharifi and J. Fan, "Self-tuning fuzzy looper control for rolling mills," *Proc. 39th IEEE Conf. on Decision and Control,* Sydney, Australia, pp. 376-381, 2000.
[6] M. T. Hagan, H. B. Demuth and M. Beale, *Neural Network Design*, Boston: PWS Pub., 1996.