The Research on Intelligent Control of AC Drive System Based on Neural Network

http://dx.doi.org/10.3991/ijoe.v9i4.2825

Gan Jia-liang,Li Zhimin,Xu Cuiqin and Tan Huaijiang
HuBei Engineering University, School of Computer and Information Science,Xiaogan, People's Republic of China

***Abstract*—Aiming at the disadvantage that traditional PID controller parameter can't be adjusted online timely, control effect is not ideal. this paper introduces a PID control algorithm based on the BP neural network by melting the regular PID controller and the self-turning learning neural network. The algorithm uses the BP neural network to build system parameter model, is able to track the changes of the controlled object, achieves a higher recognition accuracy. It amends the BP neural network weight coefficients by the BP algorithm, realizes adjustment of the PID parameters online. The results of the simulation show that these formulae has fast convergence 、high precision, and robustness.**

***Index Terms*—BP Neural Network; PID Controller; Simulation; AC Motor**

I. INTRODUCTION

The PID control is often used in the design of the AC induction motor control system due to its simple structure, easy operation some robustness error of the AC induction motor system[1] The conventional PID control hardly can achieve a good control effect because it is very tough to build a very accurate mathematical model for the AC induction motor that is a nonlinear, strong coupling of time-varying uncertain system. Therefore a speed control scheme with a good robustness must be designed to ensure the expected dynamic response. It is the hotspot of the current research to use the neural network in the identification of a dynamic system or control system due to the neural network's self-learning, self-adapting and ability to reaching any nonlinear function with any accuracy. This thesis focuses on the combination of the PID control and the BP neural network, real-time train online or batch train in proportion, integral and differential to make the PID controller got the ability of on-line regulating and optimization. The control effect of the AC vector control system of induction is eventually improved.

II. MODEL OF THE PID CONTROLLER BASED ON BP NEURAL NETWORK

2.1 Mathematical model of PID controller based BP neural network

The BP neural network that is composed by nonlinear transform units and multilayer feed forward networks has the ability of arbitrary nonlinear expression, which can get the best combination of the PID control by learning and training of the system. The structure of the PID control system based on the BP network is shown in figure 1. The controller is made up of the classic controller and the BP neural network regulator[2][3].

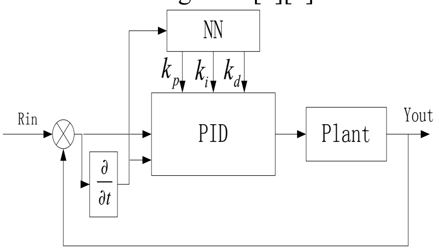

Fig 1.BP network PID controller

The classic PID controller directly controls the AC induction motor with a closed-loop control and adjusts the k_P、k_I、k_D online. The neural network real-timely adjusts the parameters of the PID controller according to status of the operation of the system, and the three outputs respectively correspond with the three parameters k_P、k_I、k_D of PID controller. The neural network automatically adjusts the k_P、k_I、k_D by the outputs after self-learning and self-setting weighting coefficient to achieve an optimal control. In digital control, the increment of PID control algorithm is shown below[1][3]:

$$u(k)=u(k-1)+k_p(error(k)-error(k-1))+k_i error(k) \\ +k_d(error(k)-2error(k-1)+error(k-2)) \tag{1}$$

The BP network has three layers and its structure is shown in figure 2:

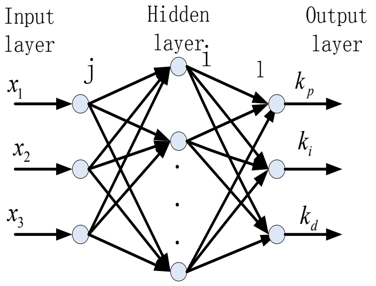

Fig 2. BP network structure

The inputs x_1、x_2、x_3 are chosen to respectively represent the error of elected asynchronous motor $error(k)$, error of variation rate $error(k)-error(k-1)$ and the derivation of error rate $error(k)-2error(k-1)+error(k\text{-}2)$. The three outputs of the output layer correspond to k_P、k_I、k_D. Input/output of hidden layer of network respectively are:

$$net_i^{(2)}(k)=\sum_{j=0}^{M}\omega_{ij}^{(2)}o_j^{(1)},$$
$$o_i^{(2)}(k)=f(net_i^{(2)}(k))\quad \text{(i=1,2L ,Q)} \qquad (2)$$

The (1), (2), (3) in above equation respectively represent input layer, hidden layer and output layer.

The activation function of the hidden neurons uses the positive symmetrical Sigmoid function, and the format is:

$$g(x)=\frac{1}{2}(1+\tanh(x))=\frac{e^x-e^{-x}}{e^x+e^{-x}} \qquad (3)$$

The input and output of the network output layer are respectively:

$$net_l^{(3)}(k)=\sum_{i=0}^{Q}\omega_{li}^{(3)}o_i^{(2)}(k) \qquad (4)$$

$$o_l^{(3)}(k)=g(net_l^{(3)}(k))\quad \text{(i=1,2,3)} \qquad (5)$$

The output is defined as: $o_1^{(3)}(k)=k_p$, $o_2^{(3)}(k)=k_i$, $o_3^{(3)}(k)=k_d$

The activation function of the output layer of neurons chooses the positive Sigmoid function because the k_P、k_I、k_D in the PID control are regularly positive[4][5][6] Its expression is:

$$g(x)=\frac{1}{2}(1+\tanh(x))=\frac{e^x}{e^x+e^{-x}} \qquad (6)$$

The real-time training algorithm is used in the design of the controller, which means that it keeps checking the difference between the setting speed $rin(k)$ of the motor and the real speed $yout(k)$ got from the real-time feedback loop to train the network all the time. The network keeps self-tuning itself and updating the parameters of the PID control to ensure that the PID controller is in a optimal parameters status during the whole process from the start to the steady brake of system. The index of the performance is[7][8]:

$$E(k)=\frac{1}{2}(rin(k)-yout(k))^2 \qquad (7)$$

The value of the network weights is modified based on the gradient descent method to minimize the $E(k)$ in the process of the back propagation of the error and the adjustment of the weighting coefficients.

$$\Delta\omega_{li}^{(3)}(k)=-\eta\frac{\partial E(k)}{\partial\omega_{li}^{(3)}}+\alpha\Delta\omega_{li}^{(3)}(k-1) \qquad (8)$$

$\alpha\Delta\omega_{li}^{(3)}(k-1)$ is the global minimum inertia term to make the search rapidly reach the convergence. η and α respectively are learning rate and inertia.

$$\frac{\partial E(k)}{\partial\omega_{li}^{(3)}}=\frac{\partial E(k)}{\partial y(k)}\frac{\partial y(k)}{\partial u(k)}\frac{\partial u(k)}{\partial o^{(3)}(k)}\frac{\partial o^{(3)}(k)}{\partial net_l^{(3)}(k)}$$
$$\frac{\partial net_l^{(3)}(k)}{\partial\omega_{li}^{(3)}(k)}\frac{\partial net_l^{(3)}(k)}{\partial\omega_{li}^{(3)}(k)}=o_i^{(2)}(k) \qquad (9)$$

Controlled object asynchronous motor is a non-linear system with strong coupled high-order multivariable so it is very tough to accurately calculate the $\frac{\partial y(k)}{\partial u(k)}$. It uses the $\mathrm{sgn}(\frac{\partial y(k)}{\partial u(k)})$ instead of $\frac{\partial y(k)}{\partial u(k)}$ make the PID control unattached to a system and got the universal applicability in simulation. At the meantime, another advantage of the change is that the vector control is not sensitive to the change of the motor parameters, different environmental and the disturbance caused by the change of the land. PID controller can automatically adjust the combination of the proportional integral and differential to get high performance control while the motor parameters greatly change (such as moment of inertia J enlarges or shrinks ten times). The inaccuracy from the change can be compensated by a proper change of the learn rate η . New equation after the above change is[6][7][8]:

$$\frac{\partial u(k)}{\partial o_1^{(3)}(k)}=error(k)-error(k-1),$$
$$\frac{\partial u(k)}{\partial o_2^{(3)}(k)}=error(k),$$
$$\frac{\partial u(k)}{\partial o_3^{(3)}(k)}=error(k)-2error(k-1)+error(k-2)$$

The learning algorithm of the output layer is easily got by the above analysis:

$$\Delta w_{li}^{(3)}(k)=\alpha\Delta w_{li}^{(3)}(k-1)+\eta\delta_l^{(3)}o_i^{(2)}(k) \qquad (10)$$
$$\delta_l^{(3)}=error(k)\mathrm{sgn}(\frac{\partial y(k)}{\partial u(k)})\frac{\partial u(k)}{\partial o_l^{(3)}(k)}g'(net_l^{(3)}(k))\quad \text{l=1,2,3.}$$

Similarly, the learning algorithm of the weight coefficients of the hidden layers is:

$$\Delta w_{ij}^{(2)}(k)=\alpha\Delta w_{ij}^{(2)}(k-1)+\eta\delta_i^{(2)}o_j^{(1)}(k) \qquad (11)$$

$$\delta_i^{(2)}=f'(net_i^{(2)}(k))\sum_{l=1}^{3}\delta_l^{(3)}w_{li}^{(3)}(k),i=1,2\text{L } Q \qquad (12)$$

In the above equation:

$g'(\bullet)=g(x)(1-g(x)),f'(\bullet)=(1-f^2(x))/2$

2.2 Adaptive learning algorithm improved by introducing momentum factor is applied to the BP network

The Standard BP neural network model includes the input layer, the hidden layer and the output layer. The layers are connected by weights and transfer function, and the learning process is composed of the forward propagation of the signal and the back propagation of the error. The method of the momentum and adaptive learning rate is used to accelerate the rate of convergence and reduce the vibration of process.

2.2.1 Momentum method

A momentum factor α is used to add a damping term to reduce the vibration of process, which makes each change of the weights added by a value that is proportional to the correction value of the weights last time. Its mathematical description is:

$$\Delta\omega_{ij}(t)=\eta(\partial E/\partial\omega_{ij})+\alpha\times\Delta\omega_{ij}(t-1) \qquad (13)$$

2.2.2 Adaptive learning rate method

The learning rate η is fixed in the standard BP algorithm. The learning rate η should be small enough to improve the prediction precision, which results in a long learning process[9]. The automatic adjustment method of learning rate is used in the training process to remove the problem. An initial learning rate firstly set and then the total error after correction is checked[10]. The learning rate will be increased if the error is going down; otherwise, the learning rate will properly be decreased. This process can be expressed as:

$$\eta = \begin{cases} \eta\times 1.05 & E(K+1)<E(K) \\ \eta\times 0.7 & E(K+1)\geq E(K) \end{cases} \qquad (14)$$

In the expression, E(k): the total error

III. THE BP NEURAL NETWORK PID CONTROLLER DESIGN

In the conclusion of the above discussion the algorithm of the PID controller based on the BP neural network can be built as follows[9][10]:

1) Set input layer nodes to 3, initialize the value of weight for every layer, learning rate η and set the inertia coefficient k to 1;

2) The values of sample $rin(k)$ and $yout(k)$ respectively correspond to the given speed of the motor and the current speed. The error of the calculation is

$error(k) = rin(k) - yout(k)$;

3) The input and output of Neural network layer of neurons NN: the output of the NN output layer is the three parameters k_P、k_I、k_D for the PID controller.

4) Calculate the output $u(k)$ of the PID controller with the following the formula

$$u(k) = u(k-1) + k_p(error(k) - error(k-1)) + k_i error(k) + k_d(error(k) - 2error(k-1) + error(k-2))$$

5) The parameters of the PID controller are automatically adjusted by the neural network learning and the online adjustment weight coefficient.

6) Set k = k + l, then go back to 1) until the error meets the requirements

IV. THE SYSTEM SIMULATION AND EXPERIMENT RESULTS

Figure 3 shows a simulation platform for the AC induction motor vector control system. The platform has four sub modules: the controller is a controller module that includes speed, current and SVPWM controllers controlled by programs and sampled every 100us; the compare is an inverter driving module that compares the input value of comparison with the triangular wave to get switch signals of the inverter; the inverter is an inverter module that gets the voltage values of the three-phase motor from the input of the switch signals, which the busbar voltage values is constant 540V; the immemo is a asynchronous motor model.

A rotor field-oriented control is built in the MATLAB/ SIMULINK. The parameters of the motor for the simulation: rating power =550KW, voltage = 350V, current = 13A and speed =1430r/min, stator resistance Rs =0.813Ω, rotor resistance Rr = 0.531Ω, excitation inductance Lm = 0.1024 H, rotor 0.1088 H, inductance Lr = 0.1063 stator inductance Ls=0.1063H, inertia J H = 0.02 kg. m^2.

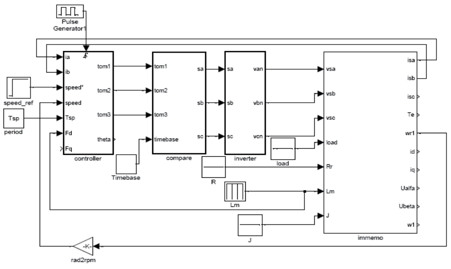

Figure 3: Simulation platform of the AC induction motor vector control system

4.1 Compare the control effect of the variable parameter motor model controlled by the BP neural network PID with by the traditional PID

When the motor in the vector control system is respectively controlled by the BP neural network PID or the traditional PID, compare the response curve of the speed with different inertia (J=0.2/0.02/0.002) and the adaptive change process of the parameters of the BP neural network control in the start-up process. The speed of the motor n is 1500rad/min and starts at 0.1s.

4.1.1 Compare the control effects when J=0.02

When J = 0.02, the control effects are respectively shown in the figure 4 and figure 5:

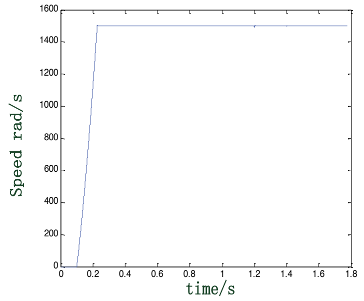

Figure 4: Control effect of traditional PID when J = 0.02

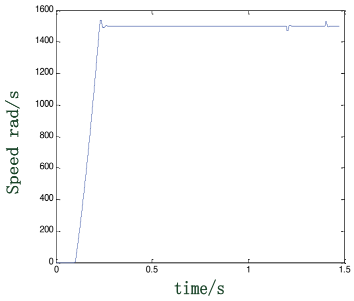

Figure 5: Control effect of the BP neural network PID when J = 0.02

The figure 4 and figure 5 show that the traditional PID controller in a particular system can get a good control effect: a small response time of system, almost no overshoot and a little of affection from the load disturbance to the system. The BP neural network PID controller can get a good control effect as well: very short rising time, a little bit of overshoot less than 3% and a low load disturbance. Though the control effect of the BP neural network PID controller is not good enough, it shows the result with high convergence speed, low overshoot and a small response to the load disturbance.

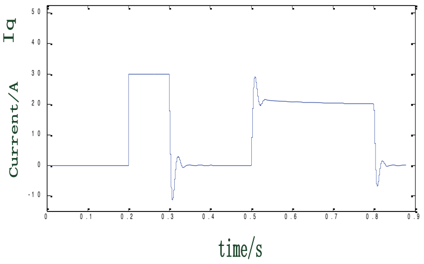

Figure 6 : Changes of the output i_q of the BP - PID controller

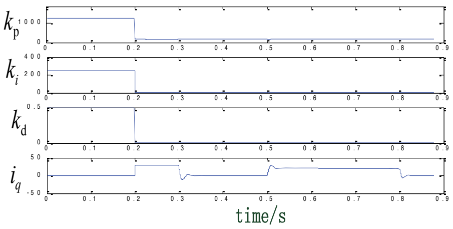

Figure 7 : Changes of k_p、 k_i、 k_d and i_q

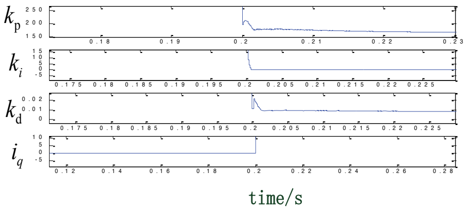

Figure 8: the changes of k_p、 k_i、 k_d at the beginning of start-up

The above figure shows that the parameters k_P、k_I、k_D change a lot at the beginning of the accelerating of the speed, at the time that the speed almost reaches the steady state and at the time when the load disturbance is applied. The parameters almost don't change during the accelerating process of the speed and after the steady state is reached.

4.1.2 Compare the control effects when J=0.2

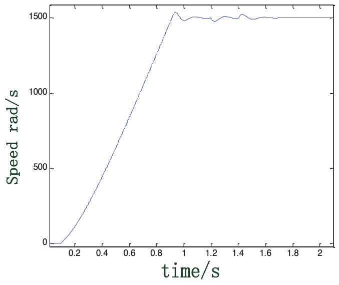

Figure 9: Control effect of the traditional PID when J = 0.2

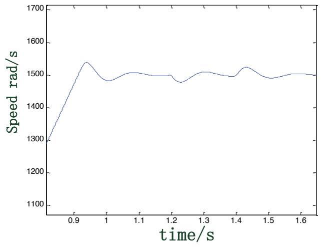

Figure 10 :The shocks after the traditional PID control reaches the steady state

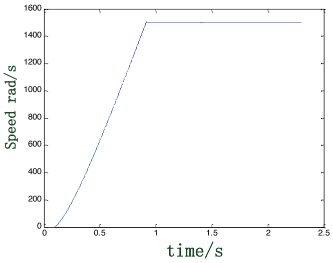

Figure 11: The control effect of the BP neural network PID at J=0.2

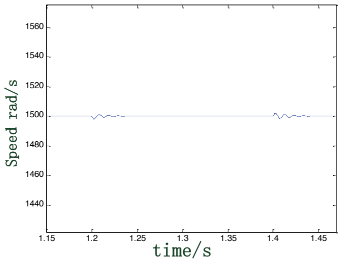

Figure 12: The control effect after the BP neural network PID reaches steady state

All the simulations use the same parameters except inertia J enlarged by 10 times. The simulation with performance at J = 0.02 doesn't reach a good enough control effect: not only is the rising time longer than the BP neural network, but also is the overshoot obvious and is the time of adjustment relatively longer. The BP network PID controller obviously has a better control effect: almost the same overshoot at J = 0.02 and higher ability of the anti-interference to the load disturbance after the steady state is reached.

4.1.3 Compare the control effects when J=0.002

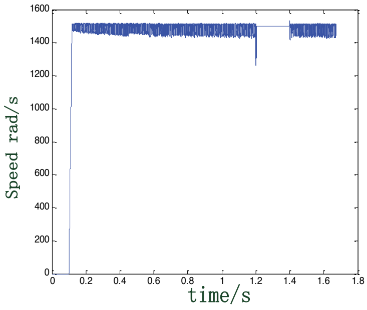

Figure 13: Control effect of the traditional PID when J = 0.002

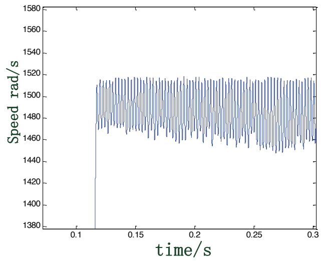

Figure 14 :The shocks after the traditional PID control reaches the steady state

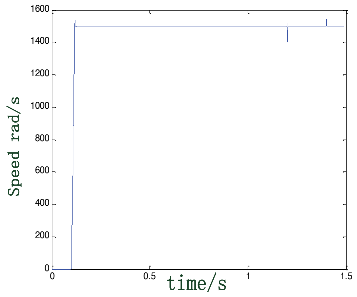

Figure 15: Control effect of the BP neural network PID when J = 0.002

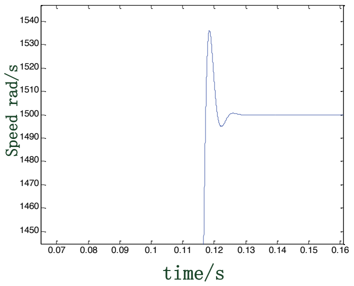

Figure 16: The control effect after the BP neural network PID reaches steady state

4.1.4.Conclusion of experiment

The result of the above simulation shows that with strong robustness, adaptability self-study ability the PID control algorithm of the BP network can dynamically change the internal structure to adapt to the changes of the environment and system. Because the gradient method uses the $\mathrm{sgn}(\frac{\partial y(k)}{\partial u(k)})$ instead of the $\frac{\partial y(k)}{\partial u(k)}$ when the BP neural network PID controller is designed, the PID controller is independent from the motor control system, in other words, the BP neural network PID for the asynchronous motor system still has high adaptability while the parameters change a lot. It can widely used in the industrial field, for example, it simply the steps to quickly change a motor without hortatory debugging at some special occasions.

4.2 Compare the control effect of the BP-PID and the traditional PID control when the motor is loaded while running

Experimental comparison as follows:

Compare the response curves of the speed during the start-up process of the motor at the different inertia (J = 0.02/0.002). The speed of the motor n = 1500rad/min. Load at 1.2s and unload at 1.4s.

4.2.1 The results of the simulation

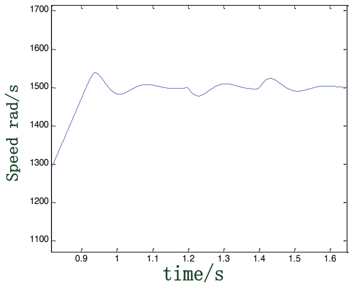

Figure 17: Anti-disturbance effects of the traditional PID at the steady state at J = 0.2

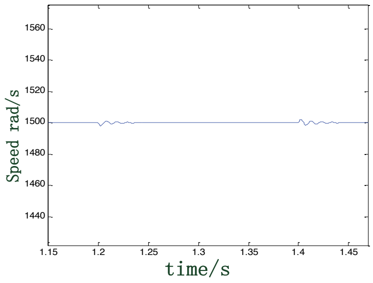

Figure 18: Anti-disturbance effects of the BP-PID at the steady state at J = 0.02

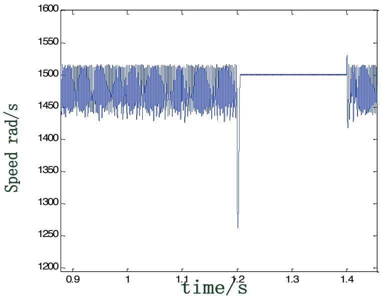

Figure 19: Anti-disturbance effects of the traditional PID at the steady state at J = 0.002

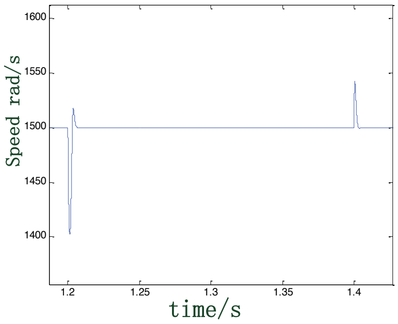

Figure 20: Anti-disturbance effects of the BP-PID at the steady state at J = 0.002

4.2.2 Results of experiment

The results of the simulation show that the algorithm of the BP network PID control has strong adaptability and high ability of anti-disturbance to the changes of the environment. It makes the system can quickly reach the steady state without a big shock while external load is applied. It is very valuable for some applications in the industrial field. It ensures the system can quickly reach a steady state without any big disturbance while there are a lot of disturbances in some special environment.

4.3 Simulation test of the BP neural network PID with the self-adaptive momentum factor

Compare the simulation result of the BP neural network PID controller improved based on the AC induction motor in the MATLAB with previous ones. The result is shown in the figure 21.

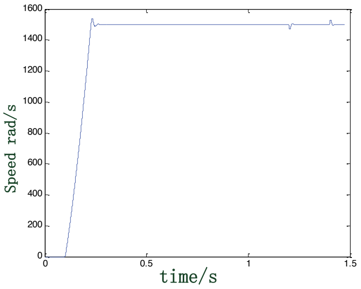

Figure 21:Simulation effect of control speed with the BP neural network PID before the improvement

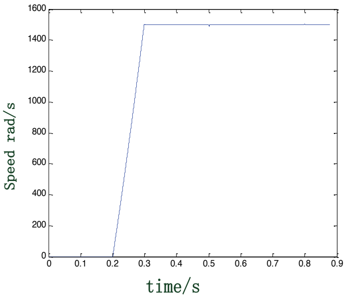

Figure 22: Simulation effect of control speed with the BP neural network PID after the improvement

The control effects in the above two diagrams indicate that the improved algorithm of control system almost doesn't have overshoot and load disturbance of the speed to get a fast response and be more sensitive as well to the change of the PID during rising process, which can respond to a tiny error in a short time. This is point can be found with the rising time and the overshoot of the simulation of the velocity loop. The output of the PID controller of the velocity loop is current i_q which value is limited in the range of [-30，30]. The i_q must be 30 during the rising process if the rise time is short, and should be quickly adjusted to minimize the overshoot while the speed is almost reaching the given value to make the system steady. The improved simulation of speed obviously shows that the system is faster and accurate to respond to the error signal.

V. CONCLUSION

The BP neural network PID controller is used to adjust the speed loop of the vector control for the AC induction motor in the simulation test. The comparison with the conventional PID control indicates that this algorithm has strong adaptability, and it can get a good control effect even if the internal parameters change a lot in the motor. It is not necessary for the algorithm to build an accurate mathematical model of the controlling object to adjust the parameters online. It have the advantages of strong nonlinear approximation ability and its robustness is higher than the conventional PID controller's.

The results of the simulations of the BP neural network PID controller and the conventional PID controller in the vector control system MATLAB show that the improved BP network PID control strategy (algorithm) is very valuable for some applications in the industrial field. It is especially useful for the system with variable parameters, strong coupling higher-order nonlinear or in a complex environment.

REFERENCES

[1] Bekir Karlik , Yousif A. Latif Al-Bastaki and S.Vakkas Ustun, 2004. Comparison Neural Networks and Ossanna Circle Diagram for Asynchronous Motors Performance Analysis.Information Technology Journal, 3: 315-318 http://dx.doi.org/10.3923/itj.2004.315.318

[2] Bose, B.K., 2002. Modern Power Electronics and AC Drives. 2nd Edn., Prentice Hall, Upper Saddle River, New Jersey, USA., ISBN-10: 013016743, pp: 736.

[3] .H.K. Ali , 2011. Fuzzy Controller Design of Servo System. Asian Journal of Applied Sciences, 4: 403-413. http://dx.doi.org/10.3923/ajaps.2011.403.413

[4] Huang Xinhan , Muhammad Arif and Wang Min , 2002. Intelligent Control: A Review. Information Technology Journal, 1: 132-135. http://dx.doi.org/10.3923/itj.2002.132.135

[5] .Ma Xiu-juan, Liu Yi, Li Ling. 2008,Research and simulation on PID control method for Brushless DC Motor based on genetic algorithm and BP neural network.[P] IEEE on Vehicle Power and Propulsion 3-5, 1~4

[6] .Shengjie, G. and L. Chunjuan, 2004. Simulation of asynchronous motor vector control system based on self-adjusting fuzzy controller. J. Lanzhou Jiaotong Univ., 343: 62-64

[7] Wei Wei,He Yan, 2005, Fundamnentals of Intelligent Control, Publishing House of Tsinghua University, Beijing.pp:60-89

[8] Wang Jian, Wang Honghua. 2008. Neural network vector control of a permanent magnet synchronous motor drive. Industrial Electronics and Applications, 3(5): 542~546

[9] Yang Chuan,Wang Xiaochun,2004,Intellectual Control System of the Induction Motor.China Journal of Scientfic Instrument, 25(3)

[10] .Zhao Jinhui, Luo Xiwen, Liu Muhua et al. 2008,Application of BP neural network to sugarcane diseased spots classification. IEEE 3: 422~425

AUTHORS

Gan Jia-liang,Li Zhimin,Xu Cuiqin and **Tan Huaijiang** are with HuBei Engineering University, School of Computer and Information Science,Xiaogan, People's Republic of Chian

This article is an extended and modified version of a paper presented at the 2013 Chinese Intelligent Automation Conference (CIAC2013), held in Yangzhou, Jiangsu Province, China, in August 2013. Submitted 25 May 2013. Published as re-submitted by the authors 23 July 2013.