Study on Neural Network Controller Based on Embedded System

Gao Shuangxi[1], Li Xi[2]

1 College of Mechanical and Electrical Engineering
Huanggang Normal University, 438000
Huanggang, Hubei Province, China
2 *School of Electrical and Information Engineering*
Wuhan Institute of Technology, 430205
Wuhan,Hubei Province, China

***Abstract* — As traditional PID controller has mature technique, it is applied widely. But the design of it depends on mathematical model of the controlled object. But the parameters of PID controller use engineering tuning method, it requires much time and effort and the parameters are only effect in a specific range. It is not suitable for complicated nonlinear and time-variable system. In recent year, people have begun to combine neural network and PID controller, and used neural network to improve traditional PID controller. The paper analyzes and studies PID controller based on neural network setting parameters and adaptive PID controller based on single neuron, and uses inverted pendulum as nonlinear object for simulation study. The paper compares and analyzes the control performance of neural network PID controller and traditional PID controller. The designed embedded neural network controller has low cost, fast response speed, good control performance and better operability and portability. The controller can be improved and designed to be a functional and practical industrial controller.**

Keywords - Single Neuron, Embedded System, Inverted Pendulum, PID Controller

I. INTRODUCTION

The design of control system is the design of regulator. And the design of regulator is the determination of regulating rule and the setting of regulator parameters. Traditional PID controller has mature technique, simple structure, good stability, reliable operation and easy adjustment, so it is widely used. But the design of PID controller depends on the experience of engineers and mathematical models of the controlled objects. Production technology of modern industry is increasingly complicated, most controlled objects have complicated nonlinearity and time-varying characteristic, and it is difficult to establish accurate mathematical model. The controlled objects are greatly influenced by noise and disturbance. Procedure parameters even model structure changes with time and working environment, which makes it difficult to determine PID controller parameters [1, 2, 3].

As a information processing technique，in recent years, neural network has developed rapidly in technological fields including intelligent control, signal processing and pattern recognition. As neural network has bionic characteristics, it can make effective use of the information of the system and is close to any nonlinear function. It has parallel processing and self-learning capacity, so it has strong fault tolerance and has self-learning and adaptive feature. Neural network has multi-input and multi-output characteristic, so it is easy to be used to control multivariable system [1, 4, 5].

But the disadvantages of neural network include slow convergence rate and long training and learning time, In order to meet the requirements of system performance, neural network controller increases the number of neurons of hidden layer, which makes the calculation amount great and makes it difficult to guarantee timeliness [6,7,8]. Therefore, there are fewer examples of neural network being applied to the control, and it is still in the stage of theory and experiment.

Inverted pendulum experiment equipment is selected as the object. The study is based on PID controller algorithm of neural network and applies good self-learning ability of neural network, which makes the parameters of the controller designed by the algorithm not only can adapt to the change of the controlled object, but also can realize real-time adaptive learning and adaptive adjustment, avoids the difficulty of establishing systematic mathematical model and designing PID parameter values, and solves the problem that it is hard for the parameters of traditional PID controller to adapt to complicated nonlinear and time variant controlled objects, for improving the performance and reliability of controller, and improving the control quality.

Constructing the hardware circuit of an embedded controller with ARM as the core needs to extend the external devices including LCD, keyboard, A/D and D/A switching circuit and storage. BOOT program, and driver of LCD, A/D device and D/A device of the embedded controller are designed, and the platform program of the controlled can be designed based on key entry scheduling corresponding application program.

PID control algorithm of neural network is transplanted into embedded system platform, which makes it become an independent and universal adaptive controller module. Storage resources are managed reasonably and graphical user interface of the controller is designed. Inverted pendulum is used as the controlled object to test and analyze the control performance of embedded PID controller of neural network.

II. PID CONTROLLER BASED ON SETTING OF BP NEURAL NETWORK

For the structure of using PID controller based on BP neural network, the common neural network includes BP neural network and RBF neural network. The chapter uses BP neural network to adjust the parameters of PID controller. PID controller directly controls the controlled object. Position-type control algorithm is used. Three parameters (kp, ki and kd) of PID controller are determined by BP neural network.

BP neural network is a multilayer feedforward neural network with hidden layers. The basic thought of the algorithm is gradient descent method which makes the actual output of the network and error mean-square value of the desired output minimal. BP neural network includes one or more hidden layers. And neurons of hidden layers use S type functions as excitation function, which can express any nonlinear relationship. Neural network of the design in the paper uses three layers of BP network, and the number of each layer of neurons is 3-5-3. The output of neurons in network output layer corresponds to three adjustable parameters of PID controller, Kp, Ki, and Kd. According to the running state of the system and performance index, neural network adjusts weighting coefficients, which makes the output of neural network correspond to the parameters of PID controller under the optimal control law.

Three outputs of output layer are x1=rin(k) , x2=yout(k) ,x3=error(k).

The input of the I node in network hidden layer is

$$net2in_i(k)=\sum_{j=1}^{4}w_{ij}x_j \qquad (1)$$

In the formula, w_{ij} is the weighting coefficient from the j node of input layer to the i node of hidden layer.

The output o f the i node is

$$net2out_i(k)=f(net2in_i(k)) \quad \text{(i=1, 2, 3, 4, 5)} \qquad (2)$$

Activation function of neurons in hidden layer takes symmetric sigmoid function $f(x)=\dfrac{e^x-e^{-x}}{e^x+e^{-x}}$.

The input of the m node in network output layer is

$$net3in_m(k)=\sum_{i=1}^{5}v_{mi}net2out_i(k) \qquad (3)$$

In the formula, v_{mi} is the weighting coefficient from the i node of hidden layerto the m node of the output layer.

The output of the m node in network output layer is

$$net3out_m(k)=g(net3in_m(k)) \quad \text{(m=1, 2, 3)} \qquad (4)$$

Three nodes in output layer corresponds to three adjustable parameters of PID controller,

$$net3out_1(k)=k_p \;,\; net3out_2(k)=k_i \;,\; net3out_3(k)=k_d \qquad (5)$$

As Kp,Ki,Kd can't be negative, activation function of neurons in output layer takes nonnegative sigmoid function $g(x)=\dfrac{e^x}{e^x+e^{-x}}$.

Learning algorithm of BP neural network takes performance index:

$$E(k)=\frac{1}{2}error(k)^2=\frac{1}{2}(rin(k)-yout(k))^2 \qquad (6)$$

Gradient descent method is used to correct weight coefficient of BP network and attaches momentum term which makes fast convergence minimal. The adjustment rule of weighting coefficient from the I node in hidden layer to the m node in output layer is as follows.

$$\Delta v_{mi}(k)=-\eta\frac{\partial E(k)}{\partial v_{mi}}+\alpha(v_{mi}(k-1)-v_{mi}(k-2)) \qquad (7)$$

In the formula, η is learning rate, and α is inertial coefficient.

$$\frac{\partial E(k)}{\partial v_{mi}}=\frac{\partial E(k)}{\partial y(k)}\frac{\partial y(k)}{\partial u(k)}\frac{\partial u(k)}{\partial net3out_m(k)}\frac{\partial net3out_m(k)}{\partial net3in_m(k)}\frac{\partial net3in_m(k)}{\partial v_{mi}(k)} \qquad (8)$$

In the formula, $\dfrac{\partial net3in_m(k)}{\partial v_{mi}(k)}=net2out_i(k)$. $\dfrac{\partial y(k)}{\partial u(k)}$ is unknown and can be replaced by sign function $\mathrm{sgn}[\dfrac{\partial y(k)}{\partial u(k)}]$. And learning rate η is used to compensate. Based on formula (5), we can get:

$$\frac{\partial u(k)}{\partial net3out_1(k)}=error(k) \quad \frac{\partial u(k)}{\partial net3out_2(k)}=\frac{T}{T_I}\sum_{j=0}^{k}error(j) \qquad (9)$$

$$\frac{\partial u(k)}{\partial net3out_3(k)}=\frac{T_D}{T}(error(k)-error(k-1))$$

So the learning algorithm of weight coefficients from network hidden layer to output layer is

$$\Delta v_{mi}(k)=\eta\delta3_m\cdot net2out_i(k)+\alpha(v_{mi}(k-1)-v_{mi}(k-2)) \qquad (10)$$

In the formula, $\delta3_m=error(k)\mathrm{sgn}[\dfrac{\partial yout(k)}{\partial u(k)}]\dfrac{\partial u(k)}{\partial net3out_m(k)}g'(net3in_m(k))$ Similarly, we can solve weight coefficients from input layer to hidden layer.

$$\Delta w_{ij}(k)=\eta\delta2_i\cdot x_j(k)+\alpha(w_{ij}(k-1)-w_{ij}(k-2)) \qquad (11)$$

In the formula, $\delta2_i=f'(net2in_i(k))\sum_{m=1}^{3}\delta3_m v_{mi}(k)$ 。

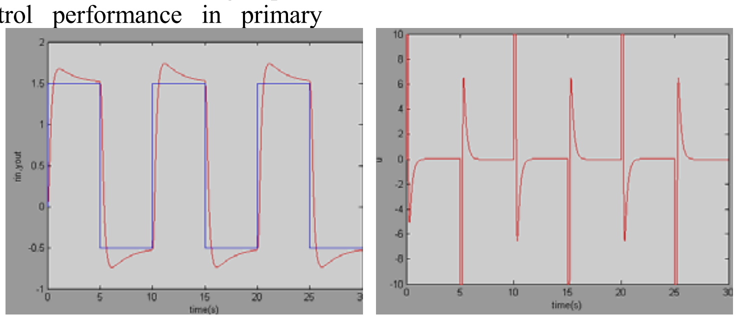
In the above formulas, $g'(\bullet)=g(x)[1-g(x)]$, $f'(\bullet)=[1-f^2(x)]/2$ 。

III. SIMULATION AND ANALYSIS OF NEURAL NETWORK CONTROL ALGORITHM

In order to make the analysis and comparison on response speed, adaptability and control effect of various algorithms easy, we can refer to square signal rin(t)=0.5+0.5*sign(sin(0.0001*2*pi*t)) and uses inverted pendulum as the controlled object. As the acceleration of actual inverted pendulum model has certain limitations, the output u of the simulated controller is limited to be + 10 ~ -10. The control on inverted pendulum is motion control which has high requirement on control precision and speed. From a large number of simulation experiments, I find that the effect of incremental PID control algorithm for inverted pendulum model is worse, so the simulation uses position-type PID control algorithm.

A. *Simulation results of traditional PID controller*

Three parameters of PID controller are a group of parameters with good control performance in primary inverted pendulum experiment devices, k_P=40, k_I=20, k_D=10. The state of inverted pendulum is controlled by the acceleration of high-speed electric machine, which uses position-type PID control algorithm. The simulation results of MATLAB are shown in Figure 1.



(a) rin and yout curve (b) Output u curve of controller

Figure 1. Simulation results of traditional PID controller

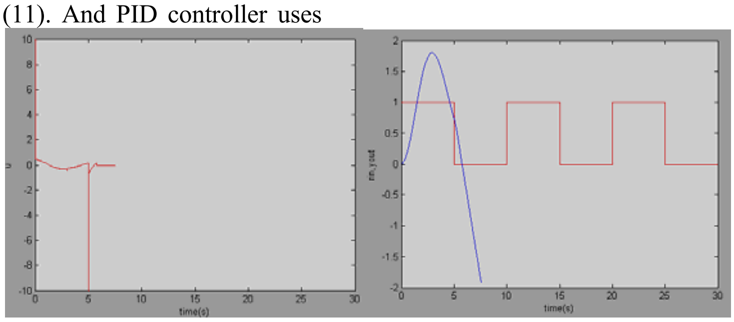
Red curve in Figure a) is the output yout of the controlled object, and blue curve is the reference input rin, for which we can see that the output of the controlled object overshoots, and the setting time is 5s nearly. In simulation program, the controller output u sets amplitude limit. From Figure (b), we can see that when reference input has tremendous changes, the controlling quantity of controller output has tremendous changes.

B. *Simulation analysis on PID controller based on setting of BP neural network*

The algorithm of simulation program is written according to the equations from (1) to (11). And PID controller uses position-type input. Before simulation, there are not trains on BP neural network. And parameter initial values are:

Learning rate η =0.2, inertia coefficient α =0.05, the initial value wi of weight coefficient matrix from the nodes in input layer to the nodes in hidden layer is generated randomly. And the initial value wo of weight coefficient matrix from the nodes in hidden layer to the nodes in input layer is generated randomly.

The simulation curve is shown in Figure 2.



(a) rin and yout curve (b) Output u curve of controller

Figure 2. Simulation results of PID controller based on setting of BP neural network

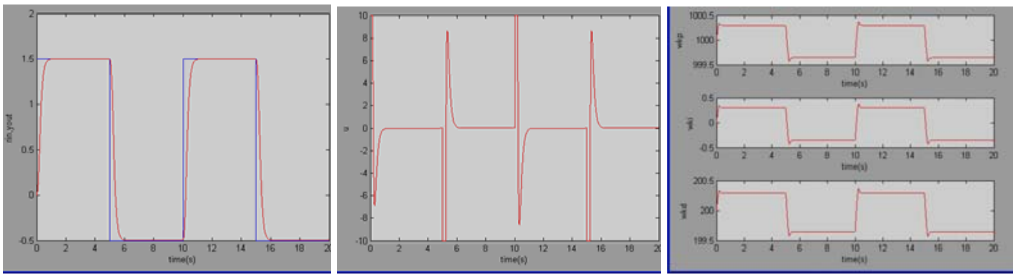
The red curve in Figure (a) is reference input rin, and the blue curve is the output state yout of the controlled object. From the simulation figure, we can see that the selected initial parameters can't satisfy the demands of the control. As there is great relationship between the performance of controller and connection weight initial value of BP neural network, the study still can't have satisfied control effect after many trials.

BP neural network algorithm has an advantage that the number of hidden layer and hidden neurons, and connection weight initial value are difficult to be determined. It is very difficult to make experiments to determine these parameters, which limits the application of it to actual control engineering [123].

C. Simulation of single-neuron PID controller based on improved and supervised hebb learning rule

The algorithm of simulation program uses position-type PID control algorithm. Learning rate is $\eta_P = \eta_I = \eta_D$ =0.0001. After several trials, the initial values of single-neuron weight (w_1 , v and w_3) are 1000, 0 and 200.

And MATLAB is used for simulation. The simulation results are shon in Figure 3.



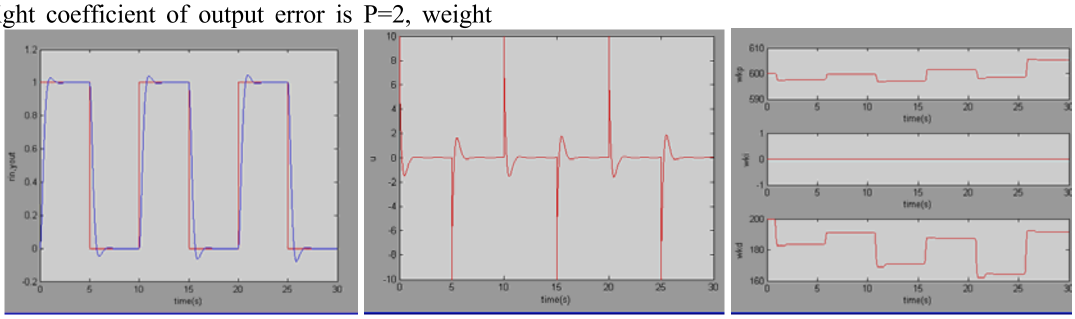
(a) rin and yout curve (b) Controlled quantity u curve (c) Variation curve of three parameters

Figure 3. Simulation results of single-neuron PID controller based on improved and supervised hebb learning rule

From Figure a, we can see that the improved control algorithm improves a lot compared with traditional PID controller, it has fast tracing speed and has no overshoot. From Figure (d), we can see that the weight of single-neuron PID controller designed by the rule is stable and has no tremendous change.

D. Simulation analysis on single-neuron PID controller based on quadratic performance index

For simulation program, proportionality factor of neurons is K=10, weight coefficient of output error is P=2, weight coefficient of controlling increment is Q=1, learning rate is η_P =0.0001， η_I =0.001 and η_D =0.0001. After several trials, the initial value of single-neuron weight w_1 , w_2 and w_3 is respectively 300, 10 and 50. The simulation results are shown in Figure 4.



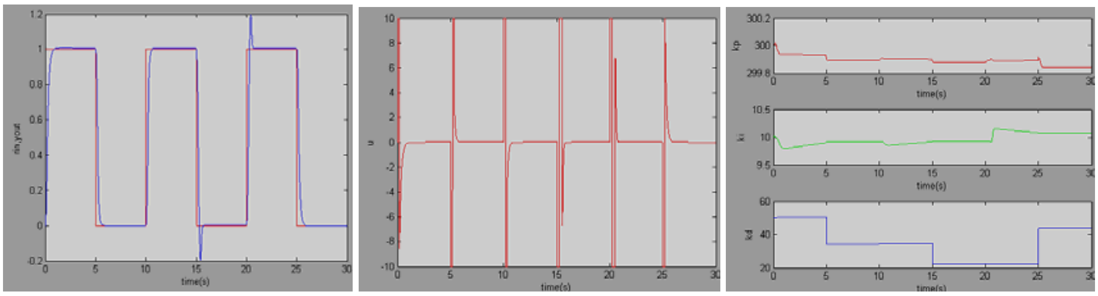
(a) rin and yout curve (b) Controlled quantity u curve (c) Variation curve of three parameters

Figure 4. Simulation results of single-neuron PID controller based on quadratic performance index

The red curve in Figure (a) is reference input rin and the blue curve is the actual state of the controlled object, yout. From Figure (a), we can see that the response speed of the controller is faster and there is overshoot. But it is smaller compared with that of traditional PID controller. And the adjustment time is very short. Figure (b) means the changing process of output u of the controller. And we can see that the performance of u improves a lot, and the situations of exceeding amplitude limit are fewer.

3.5 Simulation analysis of single-neuron PID controller based on BP neural network identification

In simulation program, after several trials, the initial values of single-neuron network weight are wcp =300， wci =10 and wck =50. In the process of NNI algorithm, the learning rate is $\eta_1 = 0.25$, the learning rate of NNC is $\eta_2 = 0.02$, and damping coefficient is β =0.01. The initial weight coefficient in output layer and input layer of NNI is generated randomly. The simulation results are shown in Figure 5.



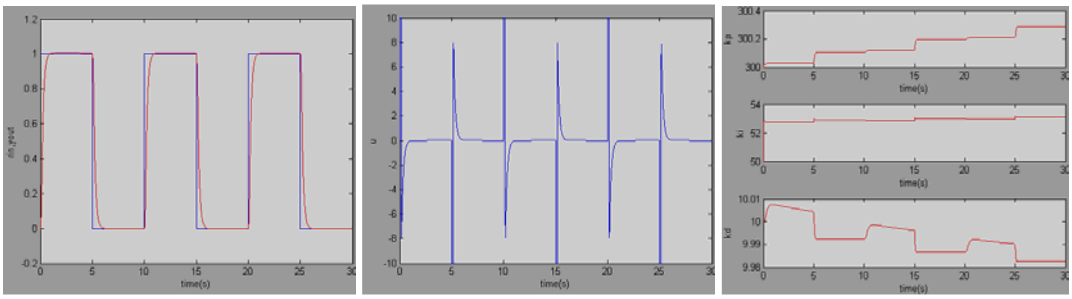
(a) rin and yout curve (b) Control quantity u curve (c)Variation curve of three parameters

Figure 5. Simulation results of single-neuron PID controller based on BP neural network identification

From Figure a, we can see that the output of the controller using the algorithm can rapidly track reference input. But when reference input changes tremendously, the output of the controlled system may have greater overshoot and the controlled quantity u changes greatly, which affects stable performance of the controlled system. Compared with single-neuron PID controller algorithm, the algorithm is complicated, has large calculation, and takes longer time to identify the controlled system.

3.6 Simulation analysis on single-neuron PID controller based on RBF neural network identification

After several trials, the initial values of single-neuron network weight are wcp=300， wci=10 and wck=50. For NNI algorithm, the learning rate is $\eta_{nni}=0.25$, the learning rate of NNC is $\eta_p=0.02$， $\eta_i=0.02$ and $\eta_d=0.01$, and the momentum factor is α =0.05. The centric vector and sound stage width of nodes in hidden layers of NNI, and the initial weight coefficients from nodes in hidden layers to nodes in output layers are generated randomly. The simulation results are shown in Figure 6.



(a) rin and yout curve (b) Control quantity u curve (c) Variation curve of three parameters

Figure 6. Simulation results of single-neuron PID controller based on RBF neural network identification

From the figure, we can see that the controller designed by the algorithm can satisfy the requirements of the control and has fast response speed. When reference input has tremendous changes, the output of the controlled object has no great transient error. Compared with sing-neuron PID controller based on quadratic performance index, the algorithm not only has large calculation, but also has slow implementation speed and response speed.

IV. ANALYSIS ON SIMULATION TEST RESULTS

After transplanting the above three controller algorithms, simulation test was made. When simulation test was made, the output of the controlled object was simulated in the program to replace the actual output of sampling objects.

Similarly, in order to make test analysis easy, reference input rin is the square signal with the cycle of 10, and the signal values are +0.5 and -0.5. In the simulation test in the chapter, the timer is not used to limit the sampling time. And the sampling time is supposed to be 0.001 seconds, so the controller needs to be sampled for 10000 times within a cycle of input signals.

You and rin are drawn on the same coordinate system. The black waveform is reference input rin, and the red waveform is the output yout of controlling objects. As the size of LCD is limited, when sample is made for 10 times, the abscissa of waveform moves to the right for a pixel. After the screen is full, it displays waveform again on the original point of abscissa axis.

For single-neuron PID controller based on quadratic performance index, the performance index is P=2, Q=1. Proportionality factor of neurons is K=10, and the learning rate is η_{kp}=0.0001, η_{ki}=0 and η_{kd}=0.0001.

For single-neuron PID controller based on RBF neural network identification, the learning rate is η_{kp}=0.02, η_{ki}=0.02, and η_{kd}=0.01, and the initial values of three parameters of PID controller are K_p =300, K_i =10 and K_d =50.

For single-neuron PID controller based on BP neural network identification, the learning rate of NNI is η_I =0.25, the learning rate of NNC network is η_c =0.02, and the initial values of three parameters of PID controller are K_p =300， K_i =10 and K_d =50. Under condition of the above initial data, the simulation test results are disappointing. The output state of the object, yout, has no poor minimum value. And the results are not consistent with the results of MATLAB simulation.

We can see form the simulation test results of two controller algorithms that the control effect is similar. But the results are achieved under the condition without considering operation time. In the paper, the experiments count the time of two algorithms completing one time of adjusting weight and figure out the time of controller output. The operation frequency of ARM controller is 200MHz.single-neuron PID controller based on quadratic performance index needs 20μs to complete one time of operation, and single-neuron PID controller based on RBF neural network identification needs 400μs to complete one time of operation, which shows that single-neuron PID controller based on quadratic performance index has smaller calculation and fast speed.

V. CONCLUSION

The paper not only studies and analyzes PID controller algorithm based on neural network setting and uses single-neuron to realize PID controller algorithm, but also makes simulation test. Experimental results indicate that using PID controller algorithm based on BP neural network setting can't achieve the desired control effect. While using BP neural network, it is difficult to determine the number of nodes in hidden layers and connection weight value, which makes the problem complicated. Using single-neuron PID controller can easily achieve the desired effect, and has ideal control effect, fast tracking velocity and strong adaptability.

In the design of embedded controller, the workload of designing graphical user interface is great. The paper sets multiple buffer zones for graphical user interface, which not only can achieve the effect of cursor and reduces the times of reading and writing NAND FLASH, but also guarantees the security of data. The design mode speeds up the velocity of the controller running. From the simulation test results, we can see that single-neuron PID controller based on RBF neural network identification can't achieve the simulation effect of the algorithm under MATLAB, which demands to adjust the parameters. Single-neuron PID controller based on quadratic performance index and single-neuron PID controller based on RBF neural network identification not only are feasible, but also have ideal simulation control effect.

ACKNOWLEDGMENT

This work is supported by the National Science Foundation of China under Grant 61175013. It is also supported by the National Science Foundation of Hubei Province under Grant 2012FFA046 and the International Cooperation Plan Project of Wuhan City under Grant 2014030709020310.

REFERENCES

[1] Bao Jian, Yu Hongming. Optimization method and application of fixed weights neural network, Computer Application, 2009, 1 , pp.230-233.

[2] Song Jinhu. Application of welding robot in China, Electric Welding Machine, 2009,4,pp.18-20.

[3] Zhu Qingzhi, Yan Baoding, Li Sujuan. Experimental study on a new AC site servo system, Micromotor, 2009,6,pp.86-88.

[4] Bai Fuming, Huang Pinwen, Zheng Lipin, Tian Lijun, Wu Ping, Ren Fazheng. Design and application of electric nose system based on ARM 9, Journal of agricultural machinery, 2009,6,pp.138-142.

[5] Li Zhi, Wang Chao, Yang Jianping, Wen Fan, Design of transformer monitor based on MCU+DSP embedded platform, Electric Power Automation Equipment, 2009,10,pp.132-135.

[6] Chen Xiaozhu, Qi Dongming, Chen Zhaozhang, Huang Yonghong, Study on biological fermentation intelligent control system based on ARM-Linux, Journal of Agricultural Mechanization Research, 2009,12,pp.158-161.

[7] Qi Dongming, Chen Zhaozhang, Zhu Xianglin, Huang Yonghong. Design and implementation of biological fermentation process intelligent control system [J], China Brewing, 2009,9,pp.102-105.

[8] Bao Jian, Zhou Bin. Study and implementation of neural network optimization method of integer weights, Computer Simulation, 2009,11,pp.195-198.

[9] Gai Bin, Liu Peng, Liu Jiafeng, Tang Xianglong. New method of extracting the characteristics of paper currency images, Journal on Communications, 2010,4,pp.128-133.

[10] Xia Wenchao, Liu Jianping, Dai Yuxing. Implementation method of neural network based on Matlab and Linu, Computer Engineering, 2010,14,pp.153-155.

[11] Lv Tao, Tang Wei, Suo Li. Short term wind speed forecasting based on chaotic phase space reconstruction theory, Power System Protection and Control, 2010,21,pp. 113-117.

[12] Hu Ligang, Xu Weiming. Study and design of embedded neural network PID controller , Computer measurement and control, 2010,9,pp.2066-2069.

[13] Wang Jianwei, Song Zhihuan. Implementation of neural network based on embedded system on training platform, Transducer and Microsystem Technologies, 2010,8,pp.100-103.

[14] Zhang Peng, Shao Huihe. Application of soft measurement based on ARM embedded system, Control Engineering, 2008,1,pp. 72-74.

[15] Jianhua Yang,wei Lu,wenqi Liu.PID Controller Based on the Artificial Neural Network. F. Yin, J. Wang, and C. Guo (Eds.): ISNN 2004, 12,pp.31-74.

[16] KOJI KASHIHARA,TORU KAWADA,KAZUNORI UEMURA,etc.Adaptive Predictive Control of Arterial Blood Pressure Based on a Neural Network during Acute Hypotension.Annals of BiomedicalEngineering,2004,32,pp.1366-1383.

[17] Tjokro S,Shah S L. Adaptive PID Control.Proceeding of the 1985 American Control Conference.1985,78,pp.1528-1534.

[18] Miller W. and Werbos P.J..Neural networks for control.MIT Press, 1990,12,pp.34-38.