A Back-stepping Neural Network Control Scheme for PM Synchronous Motors

J. Wang[1], K. M. Tsang[2], Norbert C. Cheung[2]
[1]Department of Automation, Tianjin University, P. R. China
[2]Department of Electrical Engineering, the Hong Kong Polytechnic University, Hong Kong

Abstract—**Focusing on the seriously nonlinear problem and unknown or uncertain parameters, a Backstepping control method based on Neural Networks is proposed to realize the multi-object position control of PM Synchronous Motors. Neural Networks in the scheme are used to solve the contradiction between Backstepping control and unmatched conditions of systems. A special weight online tuning method is proposed in this paper, and an off-line training phase is not required. The method does not require the system parameters to be exactly known, and the system is robust. The simulation results show that, the proposed method is effective.**

Keywords—**Neural Networks, Backstepping, PMSM Position control**

I. INTRODUCTION

The PM synchronous motor is a typical nonlinear multi-variable coupled system [5][15][16], and the control performance of PM synchronous servo motor (PMSM) drives is still influenced by the uncertainties of the load. So the nonlinear control cannot meet the requirements of its control performance. If the adaptive scheme is imposed, the parameter change and the external disturbance can be inhibited [2] [3]. But it is hard to realize and the algorithm is complex. The energy shaping can be applied to realize the nonlinear control of PMSM [13] with clear physical meaning, but there exist singular points. If backstepping [2] is imposed, the disturbance and the parameter change, which do not meet the match requirements, can be inhibited, but its realization is complicated. Nowadays the H_∞ control [19] and the intelligent control [12] [17] have also been applied to the control of PMSM. The variable structure [3] and the combination of variable structure and feedback linearization [5] are widely used in the area of PMSM control [4] to realize the global decoupling and robust control, which also achieve the PWM control directly. The neural network, which is widely and further studied [1][3], is one of the most effective schemes for the nonlinear function approximation, especially when the online learning is applied to rectify the nonlinear factors in the system and the external disturbance. [1] [3] [4] [6] are the latest literatures on neural network, in which all the current research directions are included. The control of nonlinear system, especially when the unmatched disturbance exists, is difficult to realize. [2] Proposed a Backstepping adaptive control scheme, which is a new direction in the area of nonlinear system control, and has been applied to the control of PMSM [11] [18]. However, Backstepping is suitable for addressing the problems of the linear parameter disturbance, special nonlinear parameter disturbance, and the external disturbance as well. The nonlinearity of the friction and magnetic saturations of the PMSM model cannot meet the requirements of Backstepping, but the robust control of PMSM can be achieved by applying the concept of Backstepping and the approximation characteristic of neural network.

II. MODEL OF PMSM

The model of the PMSM in the (d,q) frame can be described as:

$$\begin{aligned}
\frac{d\theta_r}{dt} &= \omega_r \\
\frac{d\omega_r}{dt} &= -\frac{B}{J}\omega_r + \frac{\tau_e}{J} - \frac{\tau_i}{J} \\
\frac{di_q}{dt} &= -\frac{R}{L}i_q - \frac{2K_t}{3L}\omega_r - i_d\omega_e + \frac{1}{L}\ u_q \\
\frac{di_d}{dt} &= -\frac{R}{L}i_d + i_q\omega_e + \frac{1}{L}u_d
\end{aligned} \qquad (1)$$

where J and B are the moment of inertia and the friction constant respectively, τ_e is electromagnet torque, which is nonlinear function of i_d, i_q and θ_r, τ_l is a load torque.

When the system parameters and the load are exactly knowable, equation (1) can be transformed into a linear system by feedback linearization and controlled by any current control scheme. However, the electromagnetic torque of the motor will introduce ripples into the system due to the nonlinearity of the permanent magnetic field and the change of the stator winding. So the linear relationship between the electromagnetic torque and the stator current cannot be held. The electromagnetic torque is shown as:

$$\frac{\tau_e}{J} = \frac{k_{t0}}{J_0}(i_q, i_d, \theta_r) + \tau(\Delta k_t, \theta) + \tau(r_s, T, \theta) \quad (2)$$

where $\tau(\Delta k_t, \theta)$ is the torque produced by the non-sinusoidal magnetic flux, and $\tau(R_s, T, \theta)$ is the dentate torque. In the same way, the variable and unknown load torque and friction coefficient can be expressed as

$$\frac{\tau_l}{J}+\frac{B}{J}\omega=\frac{\tau_{l0}}{J_0}+\frac{B_0}{J_0}\omega+\frac{\Delta\tau_l}{\Delta J}+\frac{\Delta B}{\Delta J}\omega \tag{3}$$

Then (1) can be transformed as:

$$\begin{aligned}
&\frac{d\theta}{dt}=\omega\\
&\frac{d\omega}{dt}=-\frac{B_0}{J_0}\omega+\frac{K_{t0}}{J_0}(i_q,i_d,\theta_r)-\frac{\tau_{l0}}{J_0}+\tau(\Delta k_t,\theta)+\tau(r_s,T,\theta)+\frac{\Delta\tau_l}{\Delta J}+\frac{\Delta B}{\Delta J}\omega\\
&\frac{di_q}{dt}=-\frac{R_0}{L}i_q-\frac{2K_{t0}}{3L}\omega-\Phi_0(i_d,\theta)\omega+\frac{1}{L}u_q+\frac{2\Delta K_{t0}}{3L}\omega+\Delta\Phi(i_d,\theta)\omega+\frac{\Delta R_0}{L}i_q\\
&\frac{di_d}{dt}=-\frac{R_0}{L}i_d+\Phi_0(i_q,\theta)\omega+\frac{1}{L}u_d+\frac{\Delta R_0}{L}i_d+\Delta\Phi(i_q,\theta)\omega
\end{aligned} \tag{4}$$

By setting

$\bar{x}=[x_1,x_2,x_3,x_4]^T=[\theta,\omega,i_q,i_d]^T$,

$a_1=-\frac{B_0}{J_0}$, $a_2=\frac{K_{t0}}{J_0}$, $a_3=-\frac{\tau_0}{J_0}$,

$a_4=-\frac{R_0}{L}$, $a_5=-\frac{2K_{t0}}{3L}$, $a_6=-\Phi_0(i_d,\theta)$,

$a_7=\Phi_0(i_q,\theta)$, $b=\frac{1}{L}$

$\Delta f_1(\bar{x})=\tau(\Delta k_t,\theta)+\tau(r_s,T,\theta)+\frac{\Delta\tau_l}{\Delta J}+\frac{\Delta B}{\Delta J}\omega$,

$\Delta f_2(\bar{x})=\frac{2\Delta K_{t0}}{3L}\omega+\Delta\Phi(i_d,\theta)\omega+\frac{\Delta R_0}{L}i_q$,

$\Delta f_3=\frac{\Delta R_0}{L}i_d+\Delta\Phi(i_q,\theta)\omega$

The following system can be achieved

$$\begin{aligned}
&\frac{dx_1}{dt}=x_2\\
&\frac{dx_2}{dt}=a_1x_2+a_2(x_3,x_4,\theta_r)-a_3+\Delta f_1(\bar{x})\\
&\frac{dx_3}{dt}=a_4x_3+a_5x_2+a_6x_4+bu_q+\Delta f_2(\bar{x})\\
&\frac{dx_4}{dt}=a_4x_4+a_7x_2+bu_d+\Delta f_3(\bar{x})
\end{aligned} \tag{5}$$

III. DESIGN OF CONTROLLER

Although the backstepping procedure becomes more complicated when parametric uncertainties exist in the systems, the basic idea remains the same. The complications are due to the following problems with the existing robust and adaptive procedures. First, "regression matrices" [2] in each step of the backstepping design must be determined. For example, it is well known that, in the first step of designing adaptive controllers for robots, one has to determine the regression matrix which is a very tedious and time consuming task. This procedure get sever more involved as the number of backstepping increases. Second, one basic assumption in the current robust and adaptive backstepping methods is that the unknown system parameters must satisfy the so-called linearity-in-the-parameter assumption. This may not be true in many practical situations. For example, friction in robots is a complicated nonlinear process which is hard to model as a linear-in-the-parameter process. Another example will be that certain nonlinear functions may not be linear parametrically, i.e., if there is an unknown system uncertainty.

A. NN Basics [1]

Let R denote the real numbers, R^n the real n-vectors, $R^{m\times n}$ the real $m\times n$ matrices. Let S be a compact simply connected set of R^n. With map $f:S\to R^m$, define $C^m(S)$ the functional space such that f is continuous. We denote by $\|\bullet\|$ any suitable vector norm. When it is required to be specific we denote the p-norm by $\|\bullet\|_p$. Define W as the collection of NN weights. Then the net output is

$$y=W^T\phi(x) \tag{6}$$

A general nonlinear function $f(x)\in C^m(S), x(t)\in S$, can be approximated by an NN as

$$f(x)=W^T\phi(x)+\varepsilon(x) \tag{7}$$

with $\varepsilon(x)$ a NN functional reconstruction error vector. For suitable NN approximation properties, $\phi(x)$ must satisfy some conditions.

Definition 1 [1]: Let S be a compact simply connected set of R^n, and $\phi(x): S\to R^{N_2}$ be integral and bounded. Then $\phi(x)$ is said to provide a basis for $C^m(S)$ if

1) A constant function on S can be expressed as (7) for finite N_2.

2) The functional range of NN (7) is dense $C^m(S)$ in for countable N_2.

It is emphasized that a basis $\phi(x)$ is not difficult to find. The radial basis functions, for instance, provide a universal basis for all smooth nonlinear functions [1]. [1] has shown that the approximation error in (7) can never be made smaller than order $(1/N_2)^{2/n}$ which is the order of the input space. Despite the lower bound achievable for $\varepsilon(x)$, the tracking error has been proved to be bounded, and the bound can be made small by increasing the gain of certain coefficients in the controller.

B. Controller design

From backstepping, taking system (5) as an example x_3 can be considered as the input of x_1, x_2. For PMSM, the system disturbance and the parameter change do not meet the linear parameter relationship, so the neural network is applied to approximate its nonlinear factors. Set

$$r = \dot{e} + \lambda e \tag{8}$$

$$\begin{aligned} e &= x_{1ref} - x_1 \\ \dot{e} &= \dot{x}_{1ref} - x_2 \end{aligned} \tag{9}$$

Then

$$\begin{aligned} \dot{r} &= F_1(\bar{x}) - a_2(x_3, x_4, \theta_r) \\ &= F_1(\bar{x}) - a_2 x_{3d} - a_2\eta \end{aligned} \tag{10}$$

where

$$F_1(X_1) = \ddot{x}_{1ref} - a_1 x_2 + a_3 - \Delta f_1(\bar{x}) + \lambda(\dot{x}_{1ref} - x_2) \tag{11}$$

Set $$x_{3d} = \frac{1}{a_2}[\hat{F}_1(\bar{x}) + k_1 r], \eta = x_3 - x_{3d} \tag{12}$$

Then $$\dot{r} = F_1(\bar{x}) - \hat{F}_1(\bar{x}) + a_2\eta - k_1 r \tag{13}$$

In the same way, the following expression can be held

$$\begin{aligned} \dot{\eta} &= \dot{x}_3 - \dot{x}_{3d} \\ &= F_2(\bar{x}) + a_6 x_4 + b u_q - \dot{x}_{3d} \end{aligned} \tag{14}$$

Thus $$F_2(\bar{x}) = a_4 x_3 + a_5 x_2 + \Delta f_2(\bar{x}) \tag{15}$$

Choose

$$u_q = -\frac{1}{b}[\hat{F}_2(\bar{x}) + a_6 x_4 + k_2\eta - \dot{x}_{3d} + a_2 r] \tag{16}$$

Then

$$\dot{\eta} = F_2(\bar{x}) - \hat{F}_2(\bar{x}) - k_2\eta - a_2 r \tag{17}$$

Form (5), there is

$$\begin{aligned} \dot{e}_4 &= \dot{x}_4 - \dot{x}_{4d} \\ &= a_4 x_4 + a_7 x_2 + b u_d + \Delta f_3(\bar{x}) - \dot{x}_{4d} \\ &= F_3(\bar{x}) + b u_d - \dot{x}_{4d} \end{aligned}$$

thus

$$u_d = \frac{1}{b}[-k_4 e_4 + \dot{x}_{4d} - \hat{F}_3(\bar{x})] \tag{18}$$

Define

$$\begin{aligned} F_3(\bar{x}) &= a_4 x_4 + a_7 x_2 + \Delta f_3(\bar{x}) \\ r_3 &= \hat{F}_3(\bar{x}) \end{aligned} \tag{19}$$

where $e_4 = x_4 - x_{4d}$, x_{4d} is reference value of excitation current.

In the above design process, three unknown functions $F_1(\bar{x})$ $F_2(\bar{x})$ $F_3(\bar{x})$ are presented. Generally Backstepping can be applied when $F_1(\bar{x})$, $F_2(\bar{x})$ and $F_3(\bar{x})$ meet the Linear Parameters (LP) requirement. However, it is obvious that the above $F_1(\bar{x})$, $F_2(\bar{x})$ and $F_3(\bar{x})$ do not meet the LP requirement, so the double layer neural network is imposed to approximate $F_1(\bar{x})$, $F_2(\bar{x})$ and $F_3(\bar{x})$ for the neural network holds the excellent performance of approximating nonlinear functions.

$$\begin{aligned} F_1(\bar{x}) &= W_1^T \phi_1(\bar{x}) + \mu_1 \\ F_2(\bar{x}) &= W_2^T \phi_2(\bar{x}) + \mu_2 \\ F_3(\bar{x}) &= W_3^T \phi_3(\bar{x}) + \mu_{\tau_e} \end{aligned} \tag{20}$$

where W_1^T , W_2^T , W_3^T are the ideal weighed approximation coefficients and $\phi_1(\bar{x}), \phi_2(\bar{x}), \phi_3(\bar{x})$ are the activation functions.

$$\|\mu_i\| \le \mu_{iN} \tag{21}$$

where μ_i is the approximation error, and μ_{iN} is the optimal approximation error, The estimates of $F_1(\bar{x})$, $F_2(\bar{x})$ and $F_3(\bar{x})$ are:

$$\begin{aligned} \hat{F}_1(\bar{x}) &= \hat{W}_1^T \phi_1(\bar{x}) \\ \hat{F}_2(\bar{x}) &= \hat{W}_2^T \phi_2(\bar{x}) \\ \hat{F}_3(\bar{x}) &= \hat{W}_3^T \phi_3(\bar{x}) \end{aligned} \tag{22}$$

where $\hat{W}_1^T, \hat{W}_2^T, \hat{W}_3^T$ are the estimates of W_1^T, W_2^T, W_3^T respectively, and $\widetilde{W}_1^T, \widetilde{W}_2^T, \widetilde{W}_3^T$ are estimate errors with the following relationship

$$\begin{aligned} \widetilde{W}_1^T &= W_1^T - \hat{W}_1^T \\ \widetilde{W}_2^T &= W_2^T - \hat{W}_2^T \\ \widetilde{W}_3^T &= W_3^T - \hat{W}_3^T \end{aligned} \tag{23}$$

Thus the equations of the system using the neural network control scheme can be obtained

$$\dot{r} = \widetilde{W}_1^T \phi_1(\bar{x}) - k_1 r + a_2\eta + \mu_1 \tag{24}$$

$$\dot{\eta} = \widetilde{W}_2^T \phi_2(\bar{x}) - k_2\eta - a_2 r + \mu_2 \tag{25}$$

$$\dot{e}_4 = \widetilde{W}_3^T \phi_3(\bar{x}) - k_4 e_4 + \mu_3 \tag{26}$$

and weight update law of the NN as flowing

$$\begin{aligned} \dot{\hat{W}}_1 &= \Gamma_1\phi_1(X_1) r - k_w\Gamma_1\|\varsigma\|\hat{W}_1 \\ \dot{\hat{W}}_2 &= \Gamma_2\phi_2(X_2)\eta - k_w\Gamma_2\|\varsigma\|\hat{W}_2 \\ \dot{\hat{W}}_3 &= \Gamma_3\phi_3(X_3) e_4 - k_w\Gamma_3\|\varsigma\|\hat{W}_3 \end{aligned} \tag{27}$$

where $\Gamma_1 = \Gamma_1^T > 0$, $\Gamma_2 = \Gamma_2^T > 0$, $\Gamma_3 = \Gamma_3^T > 0$, $k_w > 0$ are constant matrix.

$\varsigma = [r, \eta, e_4]^T$

IV. SIMULATION AND EXPERIMENT

The simulation use of the PMSM in a closed-loop position control on a 0.75 kW, 3000 rpm, 220v motor, the simulation parameters are:

$R = 2.875\Omega$, $\lambda_0 = 0.175Wb$, $L = 8.5e-3H$

$J = 0.6e-3kg.m^2$, $B = 0.01$, $P = 4$.

Using Gaussian function as the node function of the Neural Networks, neural network parameters: $\Gamma_1 = \Gamma_2 = 150$, $\Gamma_3 = 100$ $k_{w1} = k_{w2} = k_{w3} = 0.01$, the gains are $\lambda = 30$, $k_1 = k_2 = 400$, $k_4 = 200$. The reference position $\theta_d = 20\pi$, Simulation results are shown in figure 2. Increasing Γ_1, Γ_2 Γ_3 and decreasing k_w will shorten tuning time which is at the cost of the increasing of overshoot of ω. The significant merit of the system is that it is not necessary to know any parameters of the PMSM to implement the scheme to the congener PMSMs.

The experiment of backstepping neural network for PMSM is presented. Experiment parameters are some with simulation. The experiment results are shown in figure 3.

V. CONCLUSION

In this paper, a backstepping control scheme combined with Neural Network method for the nonlinearity of PMSM is proposed which to realize the position control. In the scheme, it is not necessary to know the dynamic parameters of the system. Compared with adaptive Backstepping control, the nonlinear characteristic of the unknown system is also unnecessarily known. The method also solved the problem of calculating complex regressive matrix in Backstepping control. In contrast with other Neural Networks control methods, the scheme does not need the off-line training stage. All the errors and weights are bounded and tracking error can be controlled to be as small as possible by selecting big enough certain gains. The simulation result verifies.

REFERENCES

[1] F. L. Lewis, S. Jagannathan and A. Yesildirek, Neural network control of robot manipulators and nonlinear systems, Taylor & Francis, Philadephia, 1999

[2] M. Krstic, I. Kanellakopoulos and P. Kokotovic, Nonlinear and adaptive control design, Wiley, New York, 1995

[3] J.T. Spooner, M. Maggiore, R. Ordonez and K.M. Passino, Stable and adaptive control and estimation for nonlinear systems: neural and fuzzy approximator techniques, Wiley Inter science, 2002

[4] K.S.Narendra and K. Partthasarathy, Identification and control of dynamical systems using neural network, IEEE, Trans. Neural Network, Vol.1, no.1,pp.4-27,1990

[5] P. A. Ioannou and J. Sun, Robust adaptive control [M], PTR Prentic-Hall, 1996

[6] Ge S.S. etc. Stable adaptive neural network control [M], Kluwer Academic Publishers, 2002

[7] R. Marino and P. Tomei, Nonlinear adaptive design: Geometric, Adaptive, and Robust, Printice-Hall, London, 1995.

[8] Vadim I. Utkin, Sliding mode control design principles and application to electric drives, IEEE Transaction on Industrial Electronics, Vol.40, No.1, pp.23-36,1993

[9] Tao and Kokotovic, Adaptive control of systems with actuator and sensor nonlinearities, New York: John Wiley&Sons, 1996

[10] Le Pioufle, Comparison of speed nonlinear control strategies for the synchronous servomotor, Electric Machine and Power systems, Vol.21, pp.151-169, 1999

[11] Jian-xin Xu, On adaptive robust back-stepping control scheme suitable for PM synchronous motors, International Journal of control, Vol.70, No.6, pp.893—920, 1998

[12] F. J. Lin, Adaptive fuzzy sliding mode control for PM synchronous motor, IEE Proceeding Control Theory and Application, Vol.145, No.1, 1998

[13] D. G. Taylor, Nonlinear control of electric machines: an overview, IEEE Control Systems Magazine, Vol.14, no.6, pp. 41-51, 1994.

[14] F.J.Lin and S.L.Chiu, Adaptive fuzzy sliding mode control for PM synchronous servo motor drives, IEE, Pro. Control Theory and Appli.Vol.145, No.1, pp.63-72, 1998.

[15] B. K. Bose, Power electronics and variable frequency drives technology and application, IEEE Press. 1997

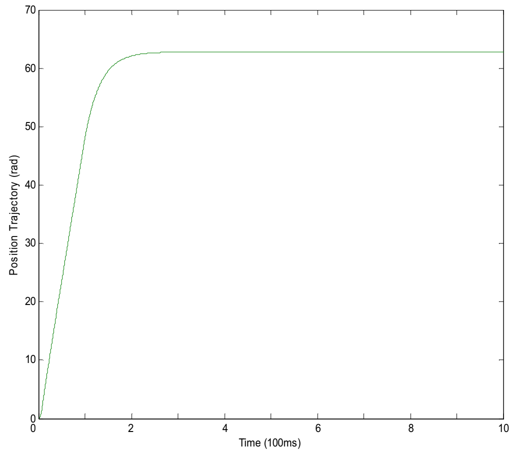
[16] P. C. Krause, O.Wasynczuk, Analysis of Electric machinery and drive system, IEEE Press, 2002.

[17] Yang Yi; Vilathgamuwa, D.M.; Rahman, M.A Implementation of an artificial-neural-network-based real-time adaptive controller for an interior permanent-magnet motor drive, IEEE Transactions on Industry Applications , Vol.39 no.1, pp.96 -104,2003

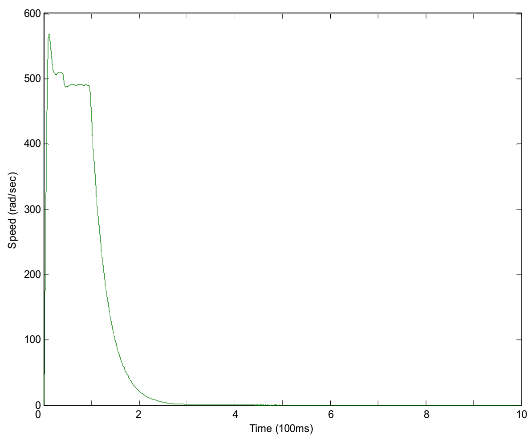
[18] Zhou, J.; Wang, Y., Adaptive backstepping speed controller design for a permanent magnet synchronous motor, IEE Proceedings-Electric Power Applications, Vol.149,no.2, pp.165 -172,2002

[19] Hsien, T.-L.; Sun, Y.Y.; Tsai, M.C., H^{∞} control for a sensorless permanent-magnet synchronous drive, IEE Proceedings-Electric Power Applications, , Vol.144,no.3, pp.173 -181,1997

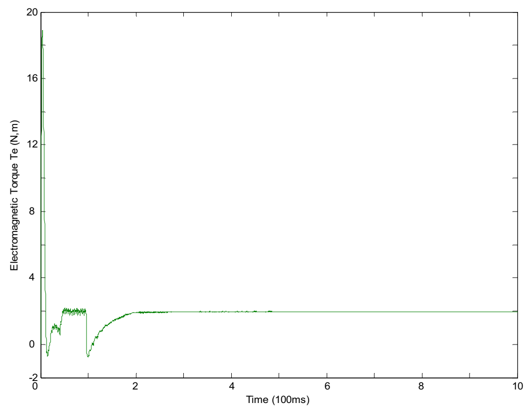
[20] Kuan-Teck Chang; Teck-Seng Low; Tong-Heng Lee, An optimal speed controller for permanent-magnet synchronous motor drives, IEEE Transactions on Industrial Electronics, Vol.41,no.5, pp.503 - 510,Oct. 1994



a, The position

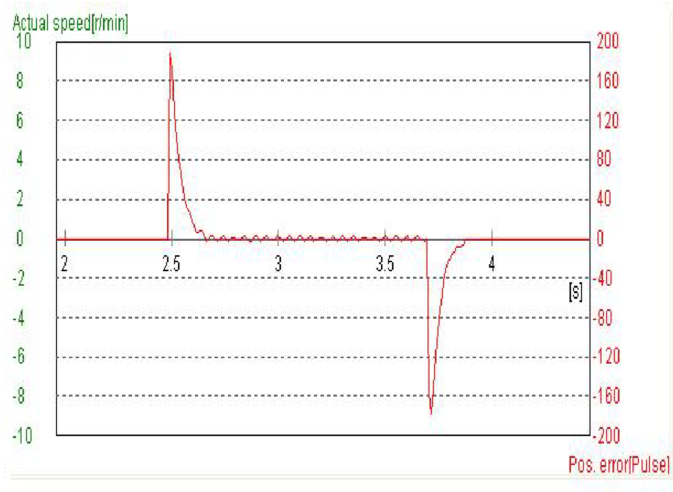


b, The speed

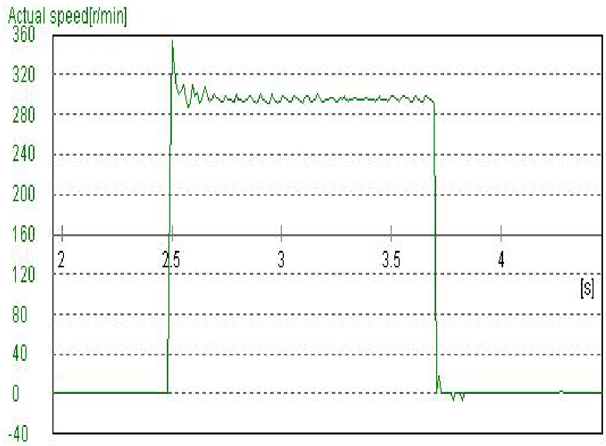


c, The torque

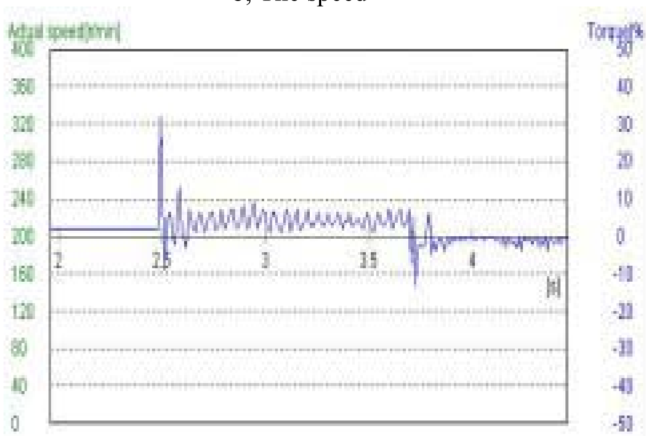
Figure 2, the simulation results



a, The position error



b, The speed



c, The torque

Figure 3 experiment results