On-line Self-tuning of PI Controller for PMSM Drives Based on the Iterative Learning Control

Yang Ming, Gao Yang, Xu Dian-Guo, Yu Yong
Dept. of Electrical Engineering, Harbin Institute of Technology Harbin, ,
Harbin, P.R.China, 150001
yangming@hit.edu.cn

Abstract—A novel on-line self-tuning of PI speed controller is given for AC Servo drive. Based on iterative learning control (ILC), the proposed scheme has solved the problem of uncertain input-output relationship. Like artificial operation, the proposed scheme can improve its control quality according to historical control experience. The feasibility and validity of the method has been proved on permanent magnet synchronous motor (PMSM) servo system.

I. INTRODUCTION

Although many novel artificial intelligence (AI) control algorithm has come forth, PID control is still used dominantly in the area of industrial drive because of its robustness and operation convenience. The conclusion in [1] has been given through widely and deeply experiment which said that the traditional PI controller is mostly better than fuzzy controller in the rate of response. But the quality of the PI controller becomes worse with the change of load torque. Then we need experienced engineer to readjust the parameter of the PID controller which is costly and unacceptable in modern industry. The self-tuning PI controller has provoked widely attention and became well developed technique especially when high quality special motion control DSP appears [2-4].

The PID self-tuning strategies can be classified into two categories: model-based method and rule-based method. As a whole, the control quality of model-based method is very good if precise model of controlled object can be deduced which needs a large amount of calculation. But actually the precise model is hard to be obtained because of nonlinear factors and noise interference. And the control quality of the system can be worse if the supposed condition in model making can not come into existence. While the rule-based method need only the input-output signal to change the parameter of the controller but not the model of the system in order to acquire satisfactory result. The process of the method is similar to the manual operation by experienced engineer. It is important and difficult to choose the self-tuning rule and the predefined regulating range of the method. The regulation process of the rule-based method is slow and time uncertain which can result in the vibration of the system. The rule-based method can be classified into two schemes:

- step-based gain tuning: using the information derived from each sampling step including the error, the error derivative and the sum of the precedent errors, etc.
- cycle-based gain tuning: using the information got from each step reference change which include overshoot and rising time, etc.

The rule-based method for PID self-tuning process has the following characters:

repeatable: the proposed output can not be acquired by only one cycle. Iterative operation is needed for a satisfactory output.

uncertain: precise mathematical equation between input (PI gain) and output (error or overshoot, etc) is hard to be given.

self-learning is expected for the process so that the knowledge in the past can be used for the following control regulation. By changing the regulation, the precise of the output is improved and the setting time becomes shorter.

Most of literatures use fuzzy rule to solve the problem of uncertainty. The quality of the control is improved with the cost of low effect more repeated operation and long setting time. This is because they ignore the cycle periodicity and self-learning ability of the process. Based on above three characters, a novel iteration self-learning scheme is designed in this paper. The proposed scheme is a rule-based cycle method which regard the PI gain as input and rise time, overshoot as output. The error of real output and expected output is used to guide the regulation of input variables. Along with the regulation process, the system can remember effective historical control experience so as to change the learning gain and speed up the time of the whole process. Figure 1 is the block diagram of self-turning control system.

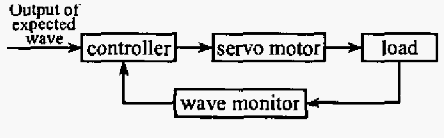

Fig.1 block diagram of self-tuning control

There are four sections of this paper. In section 2, 2.1 introduce the principle of the iterative learning control (ILC) method. 2.2 described a method to identify the moment of inertia while the self-tuning and iterate method is given in 2.3. The 3rd section is devoted to the application of the proposed

0-7803-8975-1/05/$20.00 ©2005 IEEE.

tuning algorithm and practical examples, to end the paper with section 4, which contains some concluding remarks.

II. THE PI SELF-TUNING METHOD BASED ON ITERATIVE LEARNING CONTROL

A. Principle of iterative learning control

The iterative learning control is adaptable to the controlled object which has repeating motion character. It depends not on the precise mathematics model of the controlled object, but on the antecede experience from the controller. According to the real output and expected output of the system, the controller can find the ideal control output that makes the controlled object have the expected motion[5]. During the learning process, the proposed scheme needs only to revise the control signal easily and do not need to identify the complex parameter of the system on-line as in self-adaptation control. So only a few antecede information is needed for the controller. The algorithm structure of open loop iterative learning control is shown in figure 2.

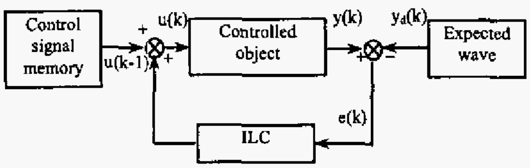

Fig.2 The structure of open loop iterative learning control

The state equation of the controlled object is as follow:

$$\begin{cases} \dot{x}(t)=f(t,x(t),u(t)) \\ y(t)=g(t,x(t))+D(t)u(t) \end{cases} \tag{1}$$

It is the commonly nonlinear equation, which has a linear output equation to input u. $x\in R^{n\times 1}$, $y\in R^{m\times 1}$, $u\in R^{r\times 1}$

f, g, D are vectors or matrixes with proper dimension.

we assume that the structure and parameter of the controlled object is unknown. The controller needs to have the output of y(t) which trace the expected output y(d) during the period $t\ni[\,t_0,\,T]$. If assume t_0=0, the discrete dynamic equation of the iterate learning scheme is as follow:

$$\begin{cases} \dot{x}_k(t)=f(t,x_k(t),u_k(t)) \\ y_k(t)=g(t,x_k(t))+D(t)u_k(t) \end{cases} \tag{2}$$

the output error is:

$$e_k(t)=y_d(t)-y_k(t) \tag{3}$$

the open loop P learning scheme is:

$$u_{k+1}(t)=u_k(t)+\Gamma(t)e_k(t) \tag{4}$$

Where $\Gamma(t)$ is the learning gain which can be linear or nonlinear function.

B. Identification of the moment of inertia

The intention to identify the moment of inertia is to make use of the information of the system. The iterative learning controller is designed to have an initial input which sufficiently approach the ideal input and ensure the stability of the initial system. The error between the above inputs can be removed gradually so that the system can have the ideal output to some extent.

For motor drive system, the current loop control is very important. The dynamic respond quality of the current loop is influenced not only by the current controller but also by stator resistance, bus voltage and some other inner factors of the servo system. It is not influenced by external mechanical parameters such as moment of inertia. So the parameters of current or torque controller are not permitted to be changed by the user in commercial AC servo controller. In addition, the frequency response of the current loop is much higher than that of the speed loop. So we can assume approximately that the input of current loop (the speed controller output) is equal to torque output of the system. The algorithm which identify the moment of inertia is based on this assumption in this paper.

The dynamic equation of PMSM system is:

$$M_e-M_l=J_{(m+l)}\frac{d\omega}{dt} \tag{5}$$

$$M_e=K\cdot I_q^{*} \tag{6}$$

M_e, M_l denote the electromagnetic torque and load torque respectively. $J_{(m+l)}$ is the sum of the moment of inertia of the motor and the load. ω is the speed of the motor; K, I_q^{*} are current-torque coefficient and current of q axis respectively.

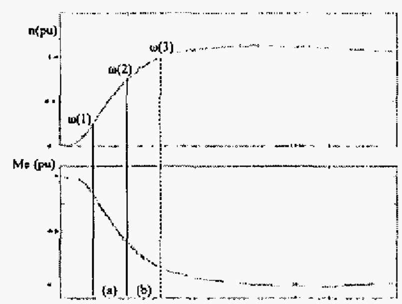

Fig.3 the test process of moment of inertia

Figure 3 shows the test process of moment of inertia. The parameters of the speed controller are set to small initial values which guarantee safe operation.

The discrete equation of (5) is as follow:

$$\bar{M}_e(a)-\bar{M}_l(a)=J_{(m+l)}\frac{\omega(2)-\omega(1)}{t_2-t_1} \tag{7}$$

$$\bar{M}_e(b)-\bar{M}_l(b)=J_{(m+l)}\frac{\omega(3)-\omega(2)}{t_3-t_2} \tag{8}$$

$\bar{M}_e$, $\bar{M}_l$ are average electromagnetic torque and average load torque during one simple time.

When

$(t2\text{-}t1)=(t3\text{-}t2)=T_{mes}$,

$\omega(2)-\omega(1)=\Delta\,\omega(a)$, $\omega(3)-\omega(2)=\Delta\,\omega(b)$

then

$$J_{(m+l)}=\frac{\bar{M}_e(b)-\bar{M}_e(a)}{\Delta\omega(b)-\Delta\omega(a)}\cdot T_{mes} \tag{9}$$

The average electromagnetic torque can be calculated by q axis current. If the load torque is constant, the calculation of moment of inertia from (9) is completely correct. If the load torque is proportional to the speed, the result of the calculation from (9) is also acceptable because the error can be neglected if the sample time is short enough which result in very small speed variation.

C. Algorithm of iterate learning control

The flow chart of self-tuning control is in Fig. 4, According to ideal model of the system, the initial gain of PI control is given corresponding to the load torque so that the initial stability of the proposed scheme can be assured.

The iterative learning control is a kind of scheme which imitate the manual operation. According to the error between actual output and real output, the PI gain is revised through the antecede experience by imitating engineer with rich experience. According to the character of PI self-tuning controller, the designed iterative learning control use open loop P learning scheme. The discrete equation of it is

$$K_p(k+1)=K_p(k)+\Delta K_p(k)$$

$$K_i(k+1)=K_i(k)+\Delta K_i(k) \tag{10}$$

where $\Delta K_p(k)$and $\Delta K_i(k)$ and are the tuning increments of $K_p(k)$ and $K_i(k)$.

In reference [6], the overshoot of the expected output is set as 7.5%, while the risetime is about 10 times of electromagnetic time constant. M_p, T_r are assumed as ideal output of the expected overshoot and risetime respectively. The discrete controller is described as:

$$\begin{cases} e_p(k)=M_p-M_p(k) \\ e_r(k)=T_r-T_r(k) \end{cases} \tag{11}$$

so

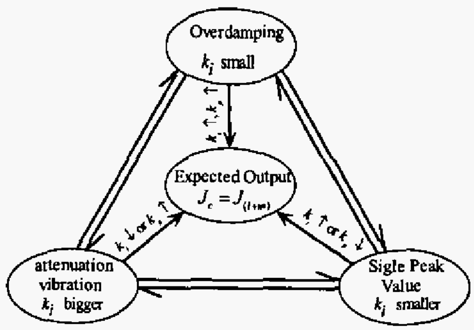

Fig.5 Relationship of Ki,Kp and output response

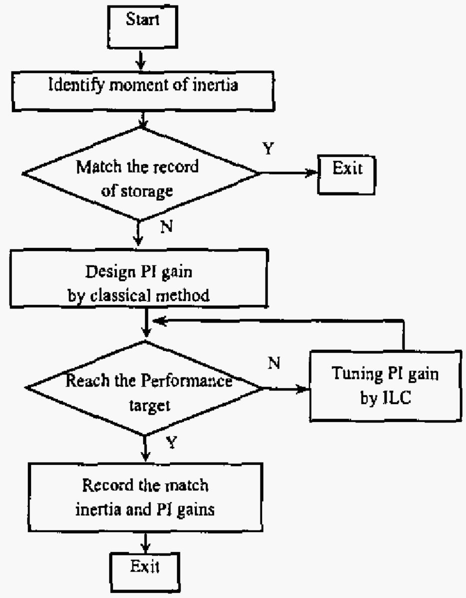

Fig.4 the flowchart of self-tuning process

$$\begin{cases} \Delta K_p(k)=\Gamma_p[e_p(k),e_r(k)] \\ \Delta K_i(k)=\Gamma_r[e_p(k),e_r(k)] \end{cases} \tag{12}$$

Γ_p, Γ_r are learning gain.

The convergence condition of learning algorithm can not be given beforehand because the limitation in iterative learning control theory. Some rules are needed to make sure the convergence of the algorithm in the proposed controller.

When overshoot appears in the system response, the change of the PI gain has little effect on risetime Tr. It means that risetime is insensitive to the change of PI gain. So the main goal of the controller is to control the overshoot Mp.

By assuming that Jc is the equivalent moment of inertia of the controller, when $Jc=J_{(l+m)}$, the expected output is reached. When Jc and $J_{(l+m)}$ is not match, the overdamping(without overshoot), single peak value(without backshoot), attenuation vibration and divergence are appeared. The process of self-tuning is to adjust Kp and Ki to make sure of $Jc=J_{(l+m)}$. Fig 5 is the relationship between parameters Kp, Ki and output response. The first three stability condition mentioned above can transform each other during the process to approach the expected output. The main principle of parameter regulation of PI gain is to make sure of the stability of the system. Once the overshoot rises, the parameter should be changed back to the primary stable one. Meanwhile, the approach strategy is also changed so that the real output approach to expected output. For example, if the system appears to be attenuation vibration, the overshoot rises when Kp increase. Then we need renew Kp and decrease Ki.

Once the expected output is reached, the system will log the current load torque and PI gain. A database of the matched parameter is also built. With the increasing time of tuning, the database information becomes rich. If the identified moment of inertia match the database information, an ideal PI gain can be acquired immediately without the adjustment process. On the other hand, the rich database can provide a large amount of historical log of effective control, which can guide the design of learning gain Γ_p ; Γ_r and improve the iterative learning process. The function of database is similar to the experience from the engineer. The learning process looks like to train an apprentice into skilled worker.

III. Results of Experiment

The construction of digital experimental system includes double TI company's DSP (TMS320LF2407A). The main drive loop uses intelligence power model (IPM). Some other interface circuits and detecting circuits are also used in the system. Some of compensation and revision methods are adopted in the system to make sure of the high performance of the servo system.

The experimental motor parameter is as follow:

rate power : 750 W
voltage : AC 200-230 V, 50/60 Hz
rate speed : 2000 rpm(torque3.6 Nm)
maximal speed : 3000 rpm
moment of inertia : 3.3e-4(Kg·m^2)

If the initial parameter respond becomes attenuate during the self-tuning process, the output reach expected value after three times of iterative operation which is shown in Fig.6. When the steady-state error exists in initial respond of the system, after the first adjustment, single peak value style appears and the output reaches expected value at the fourth iterates of auto-tuning (Fig 7). When the moment of inertia is ten times of that of the motor, the identified moment of inertia matches the value in the database as in Fig 8. So the PI gain can be acquire without iteration process. The output will reach its expected value soon.

IV. Conclusion

In this paper, a novel self-tuning scheme based on iterative learning control is designed which makes full use of the repeatability in self-tuning process of the PI controller. The uncertainty between input and output and self-learning character of the system are also considered in the proposed scheme. A simple algorithm is designed to imitate the manual process in regulating the PI gains which give a good performance of speed controller in PMSM servo system. The proposed scheme can also be used in some other area of PI gain adjustment. Because of the limitation in theory of iterative learning control, the convergence condition can not be predicted. So some rules in system respond is necessary to make sure of the convergence of the algorithm.

Fig.6 attenuation vibration response process

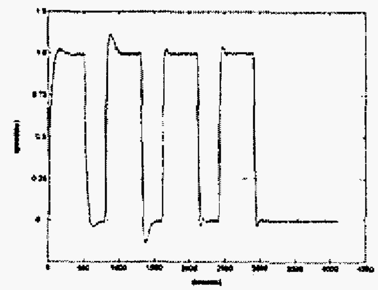

Fig.7 single peak value response process

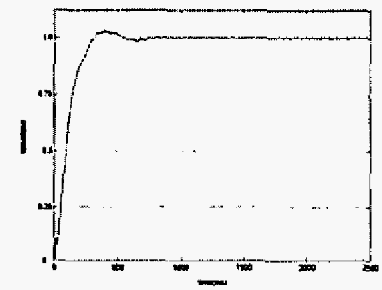

Fig.8 speed response when moment of inertia and PI gain are matched

References

[1] Zulkitilie Ibrahim, Emil Levi, *"A comparative analysis of fuzzy logic and PI speed control in high-performance AC drives using experimental approach"*. IEEE, Trans. IAS, Vol. 38,2002, pp:1210 – 1217.

[2] Bin Zhang, Yaohua Li, Yansheng Zuo, *"A DSP-based fully digital PMSM servo using on-line self-tuning PI controller"*, Power Electronics and Motion Control Conference, IPEMC 2000, Vol.2, pp:1012 – 1017.

[3] KuangYao Cheng, ShiuYung Lin, YingYu Tzou, *"On-line auto-tuning of a DSP-controlled BLDC servo drive"*, IEEE Conf. PESC. 2001 Vol.3, pp:1683 – 1688.

[4] L. Romeral, E.Aldabas, A.Arias, J.Llaquet, *"A simple self-tuning speed controller"*, IEEE, Conf. on ISIE 2000,Vol.2, pp:666 – 671.

[5] Arimoto S, Kawamura S, Miyazaki F, *Bettering operation of robotics by learing*. Journal of Robotic Systems, 1984,1(2):123 – 140.

[6] E.Chiricozzi, F.Parasiliti, M.Tursini, D.Q.Zhang, "*Fuzzy self-tuning PI control of PM synchronous motor drives*", IEEE Conf. on Power Electronics and Drive Systems, Vol.2, 1995, pp:749 – 754.