Research on Parameter Online Tuning Technology of Speed Controller of Permanent Magnet Synchronous Motor

Zhao Yibo,Zhang Huajun,Liu Dinghao
School of automation
Wuhan University of Technology
Wuhan, China
qixinhaitang369@whut.edu.cn

Pan Yu
Purchasing Department
China Nuclear Control System Engineering Co.,LTD.
Beijing, China
panyu@cncs.bj.cn

***Abstract*—For the problem of parameter self-tuning for speed controller in the vector control system of permanent magnet synchronous motor (PMSM), an online self-tuning method based on gradient search is proposed to tune the proportional and integral coefficients. The method uses the integral of the speed error square as a cost function. It calculates the gradient of cost function at each sample interval, and updates speed controller parameters according to gradient descent method. The proposed method does not need to identify the motor parameters at each iteration and it utilizes low computational resources.Because of short computation time and small memory space, the proposed method is easy to be implemented in practice. This paper simulates the performances of the speed controller implementing the proposed method under different conditions, and it compares the performances with traditional PID controller. The simulation results show that the proposed method adapts to the load change effectively and the system is stable.**

***Index Terms*—Permanent magnet synchronous motor; online self-tuning; proportional-integral controller; gradient search method**

I. Introduction

Featured by small size, light weight, and heating-free rotor, permanent magnet synchronous motor has been widely used in high-performance servo systems such as industrial robots and flexible manufacturing systems. Because the PI controller has a simple structure and is easy to implement, the permanent magnet synchronous motor often adopts a PI controller as a speed loop and a current loop controller, and because the external load fluctuation causes the system's moment of inertia to change, the speed loop controller must have a strong robustness to meet practical application requirements. In order to obtain a high-performance PI controller, the researchers proposed methods such as the Z-N method [1-2], relay feedback tuning method [3-4], Cohen-Coon method [5], critical proportioning method [6] and other methods. The Z-N tuning method is very simple and has been widely used in engineering. However, this method cannot be applied to a system that cannot oscillate critically, while multiple experiments are required to obtain the critical oscillation characteristics of the system, which may cause divergence, resulting in system breakdown [7]. The relay feedback tuning method uses the description function to approximate the process characteristics, the critical point information obtained is not accurate enough. Based on an empirical formula, the PID parameter tuning determines that the tuning result of this solution cannot have a higher quality. The disadvantage of the Cohen-Coon method is that the k_p value will be larger after tuning, and it is cumbersome to design the decay process of the system with a decay rate of 4:1 and perform the experiment [8]. The critical proportioning method has poor robustness. When the time lag of the process changes greatly, the control system will even diverge. Therefore, the control system must be adjusted to equal amplitude oscillations, which affects the normal operation of production, and unacceptable by project technician. [9]Some of the above algorithms are difficult to obtain the ideal PI parameters, and the selected literature ignores the optimization problem of the speed loop PI parameters. Therefore, it is necessary to conduct research on universal PI parameter calculation method for the permanent magnet synchronous motor speed PI controller.

In this paper, the gradient search method is used to calculate the k_p and k_i parameters. The gradient search method is an optimization algorithm, which is usually also called the steepest descent method. The method is to iteratively adjust the parameters in the opposite direction of the gradient until convergence, which is often used in machine learning and artificial intelligence to recursively approximate the minimum deviation model. This method is the most basic but important optimization algorithm in the analytical method. Gradient descent method is featured with advantages of simple principle, less workload, fewer storage variables, and low initial requirements, and optimization algorithm derived or inspired by which, such as neuron algorithms, are also widely used in the actual production process. Combining them with conventional PID control indicators can achieve better control effects [10]. This paper uses gradient search algorithm based on ISE indicator to carry out on-line self-tuning of PI controller parameters to realize the purpose of optimizing the control effect of PMSM vector control system.

978-1-7281-1312-8/18/$31.00 ©2018 IEEE

II. PMSM Vector Control Structure

The PMSM vector control strategy essentially controls the motor's direct and quadrature axis currents. According to the different requirements and structure of the servo system, the direct axis and quadrature axis current control methods are also different. The commonly used current control methods are: i_d = 0 control, power factor = 1 control, constant flux control, maximum torque current ratio control, etc. In the above scheme, the i_d = 0 control strategy is the simplest without demagnetization effect, and all of its current is used to generate torque, so the copper consumption of the motor is the least. In this paper, a convex PMSM without salient pole effect is used, and the power is low. Therefore, a current control scheme with i_d = 0 is adopted. Substituting i_d =0 into PMSM's state equation leads to the following formula:

$$\begin{bmatrix} \dot{i}_d \\ \dot{i}_q \\ \dot{\omega}_m \end{bmatrix} = \begin{bmatrix} p_n\omega_m & 0 \\ -R/L & -p_n\psi_f/L \\ \frac{3}{2}p_n\psi_f/J & 0 \end{bmatrix} \begin{bmatrix} i_q \\ \omega_m \end{bmatrix} + \begin{bmatrix} u_d/L \\ u_q/L \\ -T_L/J \end{bmatrix} \quad (1)$$

Among them, the ud and uq refer to stator direct axis and the quadrature axis voltage (V) respectively; the i_d i_q refer to motor stator direct axis and the quadrature axis current (A) respectively; ψ_f refer to the permanent magnet generated flux linkage (Wb);J is the moment of inertia ($kg \cdot m^2$),ω_m refer to the rotor's mechanical angular velocity (rad/s);T_L is the load torque ($N \cdot m$); p_n is the number of motor pole pairs; R is the resistance of the three-phase stator winding (Ω); L is the motor stator inductance (H).

According to equation (1), the vector control strategy block diagram of PMSM AC servo system shown in Figure.1 can be drawn. The system adopts a dual-loop cascaded vector control structure: the inner loop is a fast current inner one, and the outer is a speed outer loop. The dual-loop regulators all use PI regulators with simple design, good steady-state accuracy, and robustness. The regulators function as: speed loop PI regulators and current loop PI regulators.

It can be seen from the control block diagram that the performance of the speed loop PI regulator is closely related to the performance of the servo system, and its parameter tuning and optimization design are very important for improving the performance of the servo system. The speed PI controller includes proportional control and integral control. The two parameters of the proportional coefficient k_p and the integral coefficient k_i determine the dynamic and static characteristics and control performance of the PI controller. In the proportional control, the error can be quickly detected. In general, the larger the k_p value is, the stronger the control effect is and the effect of reducing the deviation can be achieved. However, relying solely on proportional control results in a static difference in the response, and the system is negatively affected by the large k_p value. In the integral control, the integral function can eliminate the static difference. As long as the deviation is not zero, the control function of the integral link will always accumulate until the error is eliminated, so the integral control can always eliminate the error to zero without limiting the adjustment time. The smaller the k_i value, the weaker the integral control function, and the smaller the overshoot, but it also slows down the system response. When the k_i value is too large, although the integral function is enhanced, the overshoot of the system response is also increased, and in severe cases, the system may even oscillate [11]. Therefore, the value of the controller parameter is very important.

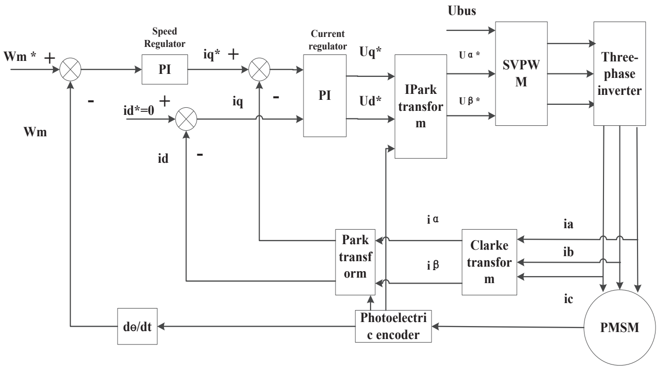


Fig. 1: Vector control diagram for PMSM AC servo system with i_d = 0

At the same time, in the actual working state, the speed and the load of PMSM cannot be constant. For the PI controller with constant parameters, it will reduce the accuracy of the motor in the process of running, and even cause unstable operation, such as the formula (2).

$$T_e - T_L = J\frac{d\omega}{dt} \quad (2)$$

The T_e is electromagnetic torque,T_L is the load torque, and J is the rotational inertia of the motor, which is the mechanical angular speed of the motor. In practical applications, the moment of inertia J will change due to the increase or decrease of the motor load, so the motor controller parameters are also needed to be appropriately adjusted. So, the constant PI controller is not suitable for high-performance permanent magnet synchronous motor vector control system. Using manual operation to debug PI controller parameters is a solution, but manual tuning takes time and effort with complex and the lowly automatic operation, which raises high requirements for theoretical training and practical experience of operators. In addition, in the process of tuning PI vector control of synchronous motor, the obtained parameter value is an optimal solution within a certain range, instead of the global optimal solution. At this time, the PI vector controller parameters need to be tuned online, to adapt to the change and improve control accuracy. Therefore, using the PI controller parameters to perform online self-tuning algorithm is the solution to the above problems.

III. On-line self-tuning algorithm for PI controller parameters

A. Objective function design

With simple structure, good control effect and high cost performance, PI controller has been widely used in the engineering field. However, when the PI controller is used to control, the k_p and k_i parameters need to be adjusted according to the mathematical model of the system. Only when the k_p and k_i parameters are within a suitable range, the entire control system will be in the best working condition, otherwise the control effect will be greatly reduced.

In the process of parameter tuning, a standard is needed to reflect the superior and inferior control effects before and after tuning. This paper mainly studies the performance of error integrals. Commonly-used single indicators are mainly used for certain characteristics of the closed-loop response of the system, such as attenuation ratio, overshoot, residual, adjustment time, and oscillation frequency. Although the characteristics of these indicators are obvious and intuitive, a control system often needs to take multiple indicators into consideration in an integrated manner. Usually, several typical deviation integration indicators are used as comprehensive performance indicators for control system response, which is as shown in the following formula (3).

$$\begin{cases} IE = \int_0^\infty e(t)dt \\ IAE = \int_0^\infty |e(t)|\, dt \\ ITAE = \int_0^\infty t\,|e(t)|\, dt \\ ISE = \int_0^\infty e^2(t)dt \end{cases} \tag{3}$$

Where $e(t)$ represents the deviation. The IE indicator in the above formula is the simplest comprehensive performance indicator based on deviation integral. However, this indicator cannot effectively present the attenuation ratio characteristics. For example, the integral of the deviation over a period of time may be zero. At this time, the IE indicator value is also zero, which cannot be directly used in actual engineering tuning. The IAE indicator takes the absolute value of the error and integrates it. When the low-order control system tune the controller parameters based on the IAE indicator, it can obtain a rapid transition and a small overshoot. The ITAE indicator is time-weighted in the expression for the deviation integral area, and it becomes more and more sensitive to long-term deviations as the time increases. That is, the same deviation increases the influence of the ITAE over time. However, the disadvantage of this kind of indicator is that the overshoot may be still larger after tuning, and the time-weighted method is more complex in program design. Compared with the above three indicators, ISE has a big difference, and the larger deviation value has a greater impact on the ISE indicator. In the expression, the ISE indicator adopts the method of integrating the square of the deviation value, which makes the ISE indicator highly sensitive to large deviations and facilitates mathematical processing. According to this characteristic, when using the ISE indicator to tune the controller parameters, there is generally not much overshoot in the response process.

In summary, IE indicator is not adopted generally; and ISE indicator is better than IAE indicator to suppress large deviation, the overshoot is small, but tuning parameters with ISE indicator may lead to unsatisfactory adjustment time; the adjustment time of system based on ITAE indicator is short, however, the amount of overshoot is large and the algorithm is more complicated. Considering the effect and feasibility comprehensively, this design chooses to tune the controller parameters of the permanent magnet synchronous motor vector control system with the ISE indicator.

B. Principles and Scheme for Implementation of Parameter Tuning Algorithm

The basic mathematical principle used by the gradient search optimization algorithm is that the negative direction of the derivative at any point on the function is the direction in which the function drops fastest at that point, that is, the direction in which the point moves toward the minimum point of the function. This principle also applies to a multivariable function. Taking any point on the objective function and seeking partial derivative for the various variables that need to be tuned, gradient vector is obtained. The gradient point is the fastest growing direction of the scalar field, so the negative direction of gradient vector is the point falls fastest. Then conduct iterative calculation of the required variables until the function value reaches a minimum value and stable, and finally get the value of the variable after tuning.

In this design, it is necessary to perform on-line self-tuning of the k_p, k_i parameters of the speed control loop PI controller of the vector control system, so that the system can obtain a better dynamic response. The tuning process is divided into sampling, tuning and output. The sampling stage uses the sampling window of the specified data length to collect the deviation in real time and calculate the deviation in the window. At the same time, the sampling window moves with the system sampling frequency as the reference, so that the window can follow the latest sampling result in real time; the tuning stage uses the data of the sampling part, and the gradient is calculated by using the gradient search method, and the controller parameters k_p, k_i are searched according to the algorithm to obtain the updated controller parameters k_p, k_i according to the algorithm in the direction of the inverse gradient; and the output stage assesses the tuning result, and input the updated controller parameter k_p and k_i to the PI controller of the speed loop. The tuning formula is as follows: The ISE indicator is

$$J = T_s \sum e^2(i) \tag{4}$$

$$e(k) = \omega^* - \omega(k) \tag{5}$$

$$\omega = u \cdot G(s) \tag{6}$$

$$u = k_p \cdot e + k_i \cdot \int edt \tag{7}$$

T_s is the sampling period of the system, ω is the real-time speed of the motor, ω^* is the specified rotational speed, $G(s)$ is the forward channel transfer function, and u is the proportional integral control function. The objective function J finds partial derivatives of k_p.

$$\begin{aligned} \frac{\partial J}{\partial k_p} &= \frac{\partial J}{\partial e} \cdot \frac{\partial e}{\partial k_p} \\ &= \frac{\partial J}{\partial e} \cdot \frac{\partial e}{\partial \omega} \cdot \frac{\partial \omega}{\partial k_p} \\ &= \frac{\partial J}{\partial e} \cdot \frac{\partial e}{\partial \omega} \cdot \frac{\partial \omega}{\partial u} \cdot \frac{\partial u}{\partial k_p} \end{aligned} \tag{8}$$

Substituting equations (4) to (7) into equation (8).

$$\frac{\partial J}{\partial k_p} = -2T_s e \sum e(k) \tag{9}$$

Set the motor parameter a constant λ, and simplify the equation (9) according to the PI controller transfer function, and obtains.

$$\frac{\partial J}{\partial k_p} = g_{k_p} = -2\lambda T_s e \sum e(i) \tag{10}$$

In the same way, we can get the partial derivative about k_i.

$$\frac{\partial J}{\partial k_i} = g_{k_i} = -2\lambda {T_s}^2 (\sum e(i))^2 \tag{11}$$

According to the iterative formula

$$k_p(k+1) = k_p(k) - \alpha_p g_{k_p} \tag{12}$$

$$k_i(k+1) = k_i(k) - \alpha_i g_{k_i} \tag{13}$$

Specific steps are as follows:

(1)Initialize parameters. Set the initial parameter value k_p and k_i, the search step length α_p and α_i, the 收敛精度 convergence accuracy R, and the data length N in the sampling window, while clear other required parameters;

(2)Collect data. The sampling window is moved according to the sampling frequency, and the deviation values in the sampling window are summed up and the error sum is output to the next part;

(3)Gradient calculations. Use equation (10) and (11) to calculate the parameter value deviation;

(4)Convergence judgment. Judge whether the demand in the third step is within the accuracy range by using the given precision R. Return to the second step if it reaches the demand, otherwise, continue the judgment;

(5)Tuning calculations. Use the equation (12) and (13) to update the controller parameters and return to step 2. Through the above steps, you can see that as long as the control system is working, the on-line parameter self-tuning of the controller will always be performed. The algorithm flow chart designed according to the above steps is shown in Figure.2 According to the above derivation formula, the most important iteration formula in the gradient search algorithm can be obtained as:

$$x_{n+1} = x_n - \alpha \cdot g_k \tag{14}$$

In (15), x denotes the parameter to be optimized, n denotes the number of iterations, a denotes the step size, and g_k denotes the partial derivative of the objective function for the variable x at the Nth iteration.

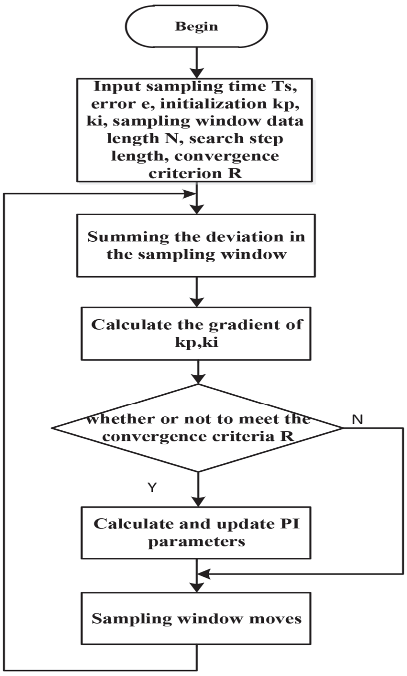


Fig. 2: Gradient method PI parameters online self-tuning flow chart

IV. IV.Simulation of Self-Tuning for Vector Control Parameters of Permanent Magnet Synchronous Motor

A. *Simulink simulation design*

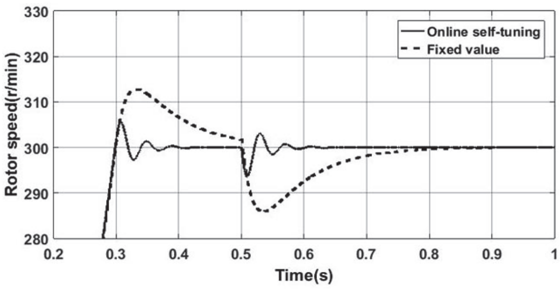
Motor parameters are shown in Table 4.1. Set the setpoint value of the speed and load torque. At t = 0s, set the speed value to 300r/min; at t = 0.5s, bring the rated load torque to 11N·m; at t = 1s, set the speed value to 500r/min; at t = 1.5s, change the load torque to -11N·m. During simulation,

TABLE I: Motor Parameters Table

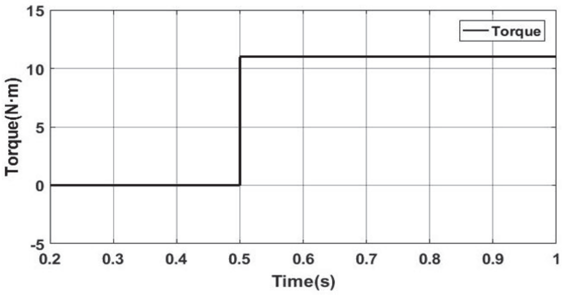
Stator resistance	D-axis inductance	Q-axis inductance
0.2	8.5e^{-3}H	8.5e^{-3}H
Number of pole pairs	Moment of inertia	Damping coefficient
4	0.089$kg \cdot m^2$	0.005$N \cdot m \cdot s$

no parameter tuning module was added. Set the k_p value to 5 and the k_i value to 40. Add the designed controller parameter online auto-tuning module and set the step size α_p to 0.03, α_i to 0.6, and the convergence accuracy R to 0.1. Perform simulation under the same input given value conditions. The resulting response is shown in Fig.3. The online self-tuning curves for k_p and k_i are shown in Fig.4.

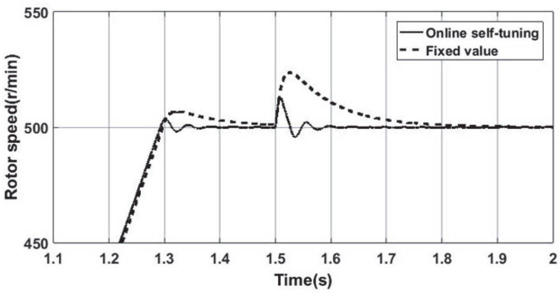
In the speed response curve, the solid line indicates the speed response curve of the online self-tuning parameter, and



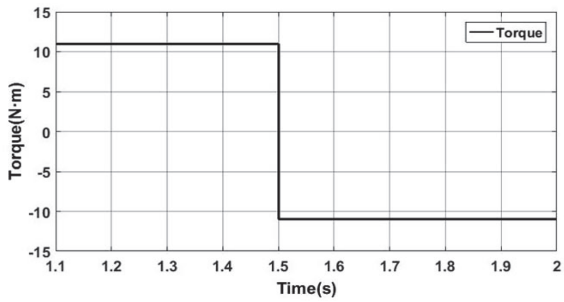
(a)



(b)

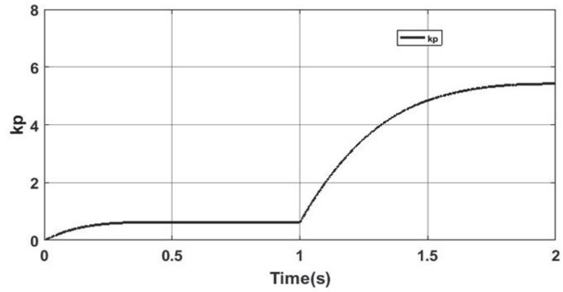


(c)

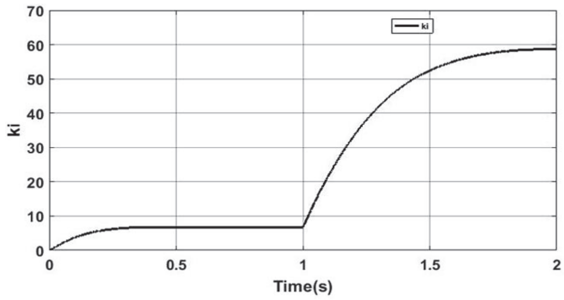


(d)

Fig. 3: Speed Response Diagram and Torque Reference Diagram (At t = 0.5s, bring the rated load torque to 11N·m; at t = 1.5s, change the load torque to -11N·m.)



(a)



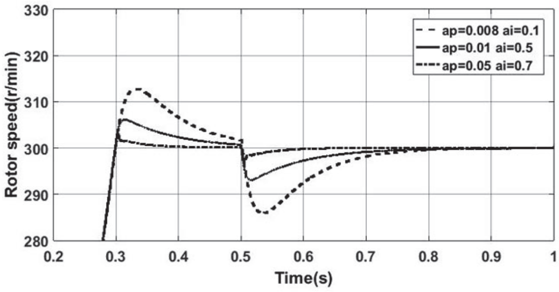
(b)

Fig. 4: Online self-tuning of k_p and k_i

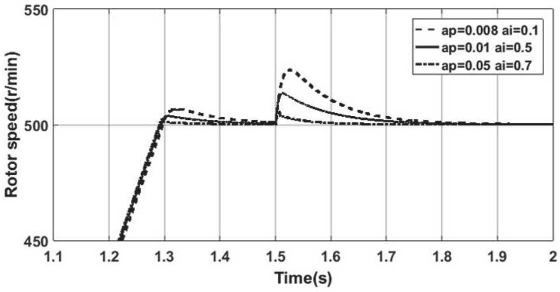
the dashed line reflects the speed response curve of the fixed parameter. From the dashed line in Figure.3, it can be clearly seen that the motor speed can roughly follow the given value, but there is a hysteresis at the start of the speed response, indicating that the proportional effect is not strong enough and there is a clear overshoot. When the torque suddenly changes, due to the presence of inertia, the speed response will show a significant delay, indicating that the proportionality coefficient is too small. These phenomena are obviously not conducive to high-performance control of permanent magnet synchronous motors. Comparing the solid line with the dashed line in Fig. 3, we find that the hysteresis is basically eliminated when the speed starts to respond, and the overshoot is also significantly reduced. When the torque suddenly changes, the motor can also quickly return to the given reference speed. However, through the analysis in Fig. 4, there are still some problems. It is found that the speed of controller parameters in the middle period of the simulation is obviously slowed down, and the responses of k_p and k_i have not fully stabilized during this period of time, which is maybe caused by too small step size and too large convergence accuracy.

According to the analysis results in Figure 3 and Figure 4, the adjusted α_p are 0.008, 0.01, and 0.05 respectively, α_i are 0.1, 0.5 and 0.7, and the convergence accuracy R was 0.01. The obtained speed response curve is shown in Figure.5.

By comparison, finally set α_p to 0.05, α_i to 0.7, and the convergence accuracy R to 0.01. The on-line self-tuning curves of k_p and k_i are shown in Fig.6. From Fig.5 and Fig.6, it can be seen that the k_p and k_i controller parameter tuning module has been working until the system is stable, and the

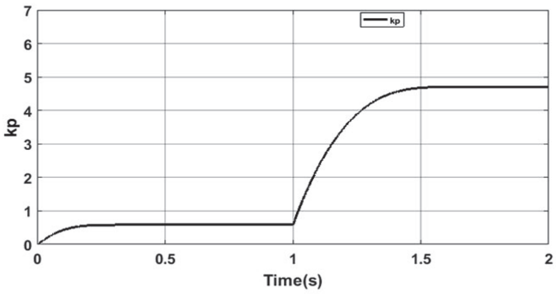


(a)

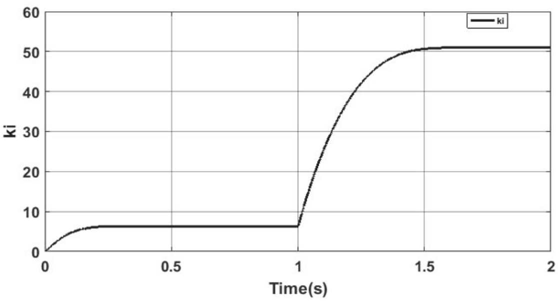


(b)

Fig. 5: Rotational speed response diagram reference diagram (At t = 0.5s, bring the rated load torque to 11N·m; at t = 1.5s, change the load torque to -11N·m.)



(a)



(b)

Fig. 6: Online self-tuning for k_p and k_i

overshoot of the speed response curve is reduced, but there is only a small static difference in the initial short period of time. This because, during the tuning algorithm, the deviation sampling window requires a certain amount of time to acquire enough deviation data. At the same time, when the load torque suddenly changes, the output speed can react quickly and the system is stable.

In summary, according to the above simulation results, it can be proved that the on-line self-tuning algorithm for speed PI parameters designed can effectively improve the control effect of the vector control system of the permanent magnet synchronous motor.

V. Conclusion

In response to the problem of parameter self-tuning for speed loop controller in the vector control system of permanent magnet synchronous motor, an online self-tuning method for proportional and integral coefficients of controller based on gradient search is proposed. By designing simulation experiments, the results show that the online self-tuning algorithm of PI controller parameters based on the ISE indicator can effectively optimize the speed loop controller PI parameters online, which can significantly improve the dynamic characteristics of the vector control system. Based on the step size, the difference in self-tuning algorithm optimization efficiency is also different. Finally, after analyzing and summarizing, the self-tuning algorithm designed achieves the desired control effect.

Acknowledgment

Thanks for the support of the foundation of MIIT, China, No. [2017]614.

References

[1] ZOU Yangyang, WANG Siyuan, TANG Changming. Research on a Self-tuning Algorithm for Improved Two-point Method without Overshoot PID [J]. Chinese Collective Economy. 2016,

[2] Liu Jun. PID Control Based on Improved Searcher Optimization Algorithm [J]. Journal of Xihua University (Natural Science). 2017,

[3] Zhan Zhanbin, Chen Feifei, Jin Bo, et al. Self-adjusting method of position control PID for electro-hydraulic proportional valve[J]. Journal of Zhejiang University (Engineering Science). 2015,

[4] Yin Huawen. Research on self-adjusting technology of temperature feedback control relay feedback parameters[J]. Automation Instrumentation. 2016,

[5] Azman AA, Rahiman M, Mohammad NN, et al. Modeling and Comparative Study of PID Ziegler Nichols (ZN) and Cohen-Coon (CC) tuning method for Multi-Tube Aluminum Sulphate Water Filter (MTAS) [M]. NEW YORK: IEEE, 2017,

[6] Wang Qinglan, Shi Xuewen, Zhang Tongyan. Critical proportionality method for tuning PID controller parameters [J]. Journal of Tangshan Teachers College. 2016,

[7] Ding Wenshuang. Self-tuning of PI parameters for permanent magnet synchronous motors [D]. 2012,

[8] Hang C C, Astrom K J, Wang Q G. Relay feedback auto-tunning of process controllers[J]. Journal of Process Control, 2002,

[9] Han Fei, Yang Chunsheng, Liu Qing. An Improved Particle Swarm Optimization Algorithm Based on Gradient Search [J]. Journal of Nanjing University. 2013,

[10] Liu Cong. Research on boiler rapid control prototyping and PID parameter self-tuning [D]. Taiyuan: Taiyuan University of Science and Technology, 2014.

[11] Yu Haisheng, Pan Songfeng, Ding Junhang, et al. Computer Control Technology [M]. Beijing: Mechanical Industry Press, 2007,