Parameter Tuning of PID Controller of Servo System Based on Particle Swarm Optimization Algorithm

Chen Lianbo¹,²,³, Xiong Xinhong³, Zheng Yousheng³, Zhang Yanhui¹,², Zhang Guoqing¹,², Wu Xinyu¹,², a, Feng Wei¹,², a
¹Guangdong Provincial Key Lab of Robotics and Intelligent System, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China
²CAS Key Laboratory of Human-Machine-Intelligence Synergic Systems, Shenzhen Institutes of Advanced Technology, Shenzhen 518055, China
³School of Logistics Engineering, Wuhan University of Technology, Wuhan 430070, China

a Corresponding author: wei.feng@siat.ac.cn; xy.wu@siat.ac.cn

Abstract. In order to improve the control precision of permanent magnet synchronous motor servo drive, a Proportional-Integral-Differential (PID) control method based on particle swarm optimization (PSO) algorithm is proposed. Firstly, the mathematical model of the permanent magnet synchronous motor was established. Then the PSO algorithm was used to adjust the parameters of the PID controller. Finally, the simulation experiment was carried out under the MATLAB/Simulink environment to test its performance. Compared with the traditional manual parameters tuning, the PID controller based on PSO algorithm had better performance and better stability.

1. Introduction

Permanent magnet synchronous motor (PMSM) has been widely used in various industrial transmission fields due to its good control characteristics, simple structure, reliable operation, small size and high power factor [1]. The control performance of the PMSM has been improved in the application of vector control. Based on vector control, the PMSM servo system often adopts the double closed loop structure of speed loop and current loop. The speed regulator and current regulator generally use proportional integral (PI) controller [2]. The regulation performance of the PI regulator is closely related to the PI parameters, while the traditional manual tuning depends on the experience and the setting effect is not guaranteed. Therefore, the tuning of PI parameters has become a research hotspot in servo systems.

In order to improve the performance of the PID controller, various scholars have proposed the parameter tuning of PID methods. In order to optimize the effect of the controller, Isidro [3] added BP neural network, and finally achieved satisfactory results. Jimenez [4] proposed a genetic algorithm to optimize the PID controller and applied the strategy to the passive optical network management. The simulation results show that compared to the Z-N tuning algorithm, the adjustment time was reduced by 64% with the above control strategy, showing a higher efficiency. At the same time, the swarm intelligence algorithm has been used to the parameter tuning of the PID controller since the rapid
development of the swarm intelligence [5]. Yan et al. [6] applied the wolf group algorithm to circuit fault diagnosis and the calculation results show that the wolf group algorithm has higher precision than the genetic algorithm. Zhao [7] proposed a particle swarm algorithm (PSO) algorithm based on multi-agent strategy for the problems of power system, and achieved good results in reactive power optimization of power system. According to the needs of medical image field, Shi [8], proposed a new algorithm for medical registration images based on PSO algorithm and hybrid mutual information. Simulation experiments show that the results of the novel algorithm are up to the sub-pixel level, and have strong robustness and adaptability.

PSO algorithm has the simple structure, strong searching ability, high computational efficiency and easy engineering implementation. It has been applied in many related fields. [9] Huang [10] used the PSO algorithm for the path planning of the manipulator, which improved the accuracy of the motion trajectory. Hu [11] applied the PSO algorithm to the hinge position of the auger mechanism of the auger, which significantly improved the performance of the luffing mechanism. Liu [12] used the PSO algorithm to optimize the air-floating frictionless cylinder structure and optimized the cylinder parameters. In order to fully solve the shortcoming of PID controller parameters of PMSM servo system, a mathematical model based on PSO algorithm for PID tuning based on PMSM is proposed. The tuning effect of the PSO algorithm on the parameters of the PI controller was been analysed with the MATLAB/Simulink.

2. Mathematical Model of PMSM based on Vector Control

The dynamic model of typical sinusoidal PMSM in the synchronously rotating reference frame is given by [13]:

$$\begin{align*}
i_d &= \frac{2}{\sqrt{3}} \left( i_s \cos \theta + i_r \cos \left( \theta - \frac{2\pi}{3} \right) + i_s \cos \left( \theta + \frac{2\pi}{3} \right) \right) \\
i_q &= \frac{2}{\sqrt{3}} \left( -i_s \sin \theta - i_r \sin \left( \theta - \frac{2\pi}{3} \right) - i_s \sin \left( \theta + \frac{2\pi}{3} \right) \right)
\end{align*}$$

(1)

$$\begin{align*}
u_d &= R_{id} + \frac{d}{dt}\psi_d - \alpha \psi_s \\
u_q &= R_{iq} + \frac{d}{dt}\psi_q - \alpha \psi_s
\end{align*}$$

(2)

$$\begin{align*}
\psi_d &= L_{dq} i_d + \psi_f \\
\psi_q &= L_{dq} i_q
\end{align*}$$

(3)

Substituting the eqn. (3) into the eqn. (2), the stator voltage equation can be obtained:

$$\begin{align*}
u_d &= R_{id} + L_{dq} \frac{d}{dt} i_d - \alpha L_s i_q \\
u_q &= R_{iq} + L_{dq} \frac{d}{dt} i_q + \alpha \left( L_d i_d + \psi_f \right)
\end{align*}$$

(4)

Where, $u_d$ and $u_q$ are the voltage along d and q axes; $i_d$ and $i_q$ are the stator current along d and q axis; $R$ is the resistance of the stator; $\psi_d$ and $\psi_q$ are the stator flux linkage along d and q axis; $\psi$ is the electrical angular velocity; $L_{dq}$ and $L_{dq}$ are the stator self-inductance in the d and q axis, respectively; $\psi_f$ represents the permanent magnet flux linkage.

In the surface-mount PMSM, the stator inductance satisfies $L_{dq}=L_{dq}=L_s$, so the electromagnetic torque equation is:

$$T_e = \frac{3}{2} p a \psi f$$

(5)
In the double closed loop control system with speed loop and the current loop, the actual current of the motor is equal to the given current through the fast current control, so the two current components can be controlled separately. Figure 1 is the schematic diagram of PMSM vector control when using id=0. The main performance of the system is determined by the speed loop, and the quality of the speed loop will affect the performance of the position loop. Therefore, the speed loop PI controller parameters are mainly adjusted in the controller parameter tuning.

![Figure 1. PMSM Vector Control Schematic](image)

3. Particle Swarm Optimization Algorithm

3.1. The Basic Principle of Particle Swarm Optimization Algorithm

Suppose a group of M particles flies at a certain speed in the D-dimensional space. The state attribute of particle i at time t can be expressed as:

Position:

\[ x_i^t = (x_{i1}^t, x_{i2}^t, ..., x_{id}^t, ..., x_{iD}^t)^T \]  

(6)

Speed:

\[ v_i^t = (v_{i1}^t, v_{i2}^t, ..., v_{id}^t, ..., v_{iD}^t)^T \]  

(7)

Individual optimal position:

\[ p_i^t = (p_{i1}^t, p_{i2}^t, ..., p_{id}^t)^T \]  

(8)

Global optimal position:

\[ p_i^t = (p_{i1}^t, p_{i2}^t, ..., p_{id}^t)^T \]  

(9)

According to the above information, the speed of particle i at time t+1 can be expressed as:

\[ v_{id}^{t+1} = w v_{id}^t + c_1 r_1 (p_{id}^t - x_{id}^t) + c_2 r_2 (p_i^t - x_{id}^t) \]  

(10)

\[ x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1} \]  

(11)
Where: \( w \) is the inertia weight; \( r_1, r_2 \) are random numbers uniformly distributed in the interval \((0, 1)\); \( c_1, c_2 \) are learning factors.

The eqn. (10) is mainly composed of three parts: the first part is the last particle velocity with inertia weight, which is the inheritance of the particle to the previous state, the particle's trust to the spontaneous state of motion, and the inertial motion according to its own speed; the second part is the "cognitive" part, it is expressed as the thinking of the particle itself, that is, the particle movement is influenced by the experience of its own movement, so that the particle has a strong local search ability; the third part is the "social" part, indicating that the movement of the particle is affected by the whole. The influence of particle swarms, the information sharing between particles and mutual cooperation.

3.2. Algorithm Implementation Flow

In the optimization process of PSO algorithm, an evaluation function is proposed to calculate the fitness value of particle. The fitness value is used to evaluate the performance of particle. The product of the absolute value of error and time is used to the test function in this paper. The expression is as Eqn. 6:

\[
F = \int_0^t t |e(t)| dt
\]  

(12)

The process of PSO algorithm is as follows:
1. Produce particle swarms and initialize particle parameters.
2. Calculate the fitness value of the particle according to eqn. (12), and determine the individual optimal value \( F_{\text{best}} \) of the particle and the global optimal value \( F_{\text{global}} \) of the particle swarm.
3. For each particle, compare the fitness value of the historical optimal position with the current particle, and update the optimal position of the individual according to the result.
4. For each particle, compare the fitness value of the global optimal position of the particle group with the fitness value of the current particle, and update the global optimal position according to the result.
5. Update the position and speed of particles.
6. Determine whether the termination condition is currently reached. If reached, the iteration is terminated; otherwise return to 2. The condition for terminating the iteration is that the number of iterations reaches the maximum number of times or the fitness value is less than the convergence precision.

4. Design of Speed Loop PI Controller Based on Particle Swarm Optimization algorithm

The schematic of the PI control system based on PSO algorithm is shown in Figure 2. In the design of the PI regulator, the key is to determine the reasonable values of \( K_{\text{pw}} \) and \( K_{\text{tw}} \). The system has better performance after adjustment by the PI controller. The range of values of \( K_{\text{pw}} \) and \( K_{\text{tw}} \) is determined by empirical methods and PSO algorithm is adopted to find the optimal value in this interval.

![Figure 2. PID Control Schematic Based on PSO](image)

A schematic diagram of the process of tuning the velocity loop PID parameters by using the PSO algorithm is shown in Figure 3.
The specific process is presented as follows:

1. A particle swarm with a population size of N=30 is generated within the range of the controller PI parameter, and each parameter of the particle swarm algorithm is initialized. Such as inertia weight, learning factor and number of iterations.

2. In the particle swarm, the position of the particle represents the parameter that needs to be optimized (K_{pw} and K_{iw} in the PI parameter of the velocity loop). According to the range of K_{pw} and K_{iw} value adjustment, the maximum position X\text{max} and the minimum position X\text{min} of the particle position are set, and the maximum velocity V_{max} of the particle is determined thereby.

3. The position information x_i and the velocity information v_i of the particles in the particle group are initialized. And calculate the fitness value F of the particle according to formula (12), set the current position of the particle as the best g\text{best} of the individual, and record the best position of all the particles as the global optimal z\text{best}.

4. For each particle, the fitness value of the particle is calculated according to equation (12). If the current particle's fitness value F_i is less than F\text{best}, it indicates that the current position is better than g\text{best}. Assign x_i, a g\text{best} to the latest individual optimal value to complete the individual's best update.

5. Comparing the fitness value of the current particle with the fitness value of the global optimal z\text{best}, if the current particle's fitness value F_i is less than F\text{best}, it indicates that the current position is better than z\text{best}. Assign x_i, a z\text{best} to make it the latest global optimal and complete the global best update.

6. The position and velocity of the particles are updated according to equations (10) and (11). The PI value in the updated particle is substituted into the controller, the system is run, and the corresponding fitness function is calculated.

7. After repeated iterations, it is determined whether the termination condition is currently reached. If the termination condition is reached, the iteration is terminated; otherwise, the loop iteration is continued. The condition for terminating the iteration is that the number of iterations reaches the maximum number of times or the fitness value is less than the convergence precision.

When the PSO algorithm ends the optimization, the optimal result is output. Comparing the optimization results with the performance of the current PI parameters, if it is better than the PI parameters before optimization, the original PI parameters are replaced to achieve the purpose of parameter self-tuning.

5. Comparison of Manual Parameters Tuning and Simulation Analysis Based on PSO

According to the vector control principle of PMSM controller, the parameter tuning simulation model of PMSM controller based on PSO algorithm is explore under MATLAB/Simulink environment. The
Simulation model is shown in Figure 4. The vector control and adaptive value calculation of PMSM is done in Simulink, while the PSO algorithm is implemented in MATLAB.

![Figure 4. PMSM Simulation Model Based on PSO](image)

After 50 iterations of the PSO algorithm, the parameters of the speed loop PI controller self-tuning parameters $K_{pw}$ and $K_{iw}$ are 0.18 and 3.95 respectively. The servo system PI controller is repeatedly debugged by traditional manual, and the best parameters $K_{pw}$ and $K_{iw}$ are 0.14 and 7. The speed response curves corresponding to two different parameters of the PI controller in two cases is shown in Figure 5.

![Figure 5. Servo System Speed Response Curve](image)

As can be seen from Fig. 5, the PI controller based on the particle swarm optimization algorithm has a significant reduction in the overshoot of the system, although the response time is not greatly improved compared to the traditional manual parameters tuning method.

In order to describe the difference between the two methods, the performance of the PSO algorithm and the manual parameters tuning are compared from the three aspects of overshoot, rise time and product integral of error and time (ITEA). The overshoot reflects the degree to which the output value deviates from the stable value; the rise time reflects the time required to stabilize the control process area; the ITAE reflects the overall stability of the control process, and the comparison results are shown in Table 1.
Table 1. Permanent Magnet Synchronous Motor Parameters

| Setting method | Overshoot/% | Rise/s | ITEA |
|----------------|-------------|--------|------|
| Trial method   | 15.8        | 0.045  | 0.071|
| PSO            | 6.6         | 0.04   | 0.040|

As shown in Table 1, the PSO algorithm has improved over the traditional manual parameters tuning in terms of overshoot, rise time and ITEA.

6. Conclusion

Based on PSO algorithm, an intelligent PID tuning optimization algorithm is proposed. After theoretical analysis and simulation experiments, the PSO algorithm could improve the tuning efficiency of the parameters compared with the traditional manual parameters tuning method. The effectiveness, overshoot, response speed and feasibility of this algorithm have been verified. The dynamic performance and steady state performance of the servo system of the motor has greatly improved.

Acknowledgements

This work is financially supported by Guangdong frontier and key technological innovation (No.2017B090910013), Shenzhen knowledge innovation technology basic research project (JCYJ20170818153048647), National Natural Science Foundation of Guangdong (No.2016A030313177), Shenzhen Technology Project (JSGG20160301160759264, JSGG20171823140127645).

References

[1] Zhang H Y, Xu H P, Fang C and Xiong C 2018 Transactions of China Electrotechnical Society 33 2923-2934 (2018)
[2] Jan R M, Tseng C S, Liu R J Electric Power Systems Research 78 1161-1168 (2008)
[3] Isidro S, Julio R B and Antonio A A Journal of Food Engineering 46 21-29 (2000)
[4] Jimenez T, Merayo N 2015 Expert Systems With Application 42 9211-9229 (2000)
[5] GirirajKumar S M, Jayaraj D and Kishan A R International Journal of Computer Applications 19 12-18 (2010)
[6] Yan X L, Ding P and Ma J Computer Engineering and Applications 53 152-156 (2017)
[7] Zhao B, Cao Y J Proceeding of the CSEE 25 1-7 (2005)
[8] Shi Y X, Qiu T S, Han J and Jin Sheng Chinese Journal of Biomedical Engineering 34 1-7 (2015)
[9] Zhang G H, Shao Xin Y, Li P G and Gao L Computers & Industrial Engineering 56 1309-1318(2009)
[10] Huang Q J, Zhang K, Song J C and Yu Z L Journal of Northeastern University 11 1636-1641 (2018)
[11] Hu J P, Peng Y M Chinese Journal of Engineering Design 5 561-566 (2018)
[12] Lu Y, Zhao G X Journal of System Simulation 9 3564-3570 (2018)
[13] Kiran B, Kottulil A M and Ananthamoothy N P International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering 1 689-695 (2013)