Design of PI Controller for PMSM using Chaos Particle Swarm Optimization Algorithm

Zuo Ru1,2,3,4, Xiong Xinhong1,*, Chen Lianbo1, Guo Shifeng2,3,4,* , Zhang Yanhui1,2,3,4, Li Daoqi1,2,3,4 and Feng Wei2,3,4,*

1School of Logistics Engineering, Wuhan University of Technology, Wuhan 430070, China
2Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China
3Guangdong Provincial Key Lab of Robotics and Intelligent System, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China
4CAS Key Laboratory of Human-Machine-Intelligence Synergic Systems, Shenzhen Institutes of Advanced Technology, Shenzhen 518055, China

Corresponding author’s E-mail contact: wei.feng@siat.ac.cn; xiongxh@whut.edu.cn; sf.guo@siat.ac.cn

Abstract. The proportional-integral-derivative (PID) controller parameters tuning is critical for improving the performance of permanent magnet synchronous motor (PMSM) systems. To improve the performance of particle swarm optimization (PSO) in the PID parameters tuning, the chaos particle swarm optimization (CPSO) is adopted to tune PI parameters in this work. The PMSM vector control system is selected as the research object, and its speed loop PI controller using CPSO algorithm is implemented. The optimal PI controller parameters are obtained using the CPSO algorithm with ten iterations. The simulation is conducted using MATLAB/Simulink to evaluate the PI controller performance. Four performance indicators including overshoot, rise time, peak time and ITAE are used to evaluate the PI controller with CPSO, and the results suggest that the CPSO algorithm has better performance and stability compared with other tuning methods.

1. Introduction

The permanent magnet synchronous motor (PMSM) has the advantages of high efficiency, high power density, high torque inertia ratio, low noise, and maintenance-free [1], and has been widely used in electric vehicles, aerospace and robotics etc. Its control research is therefore receiving increasing attention [2]. Since its introduction, the PID controller has been widely used in various fields of modern industry because of its simple structure and excellent performance. Considering the performance of the PID controller is closely related to the PID parameters [3], it is therefore critical to study the tuning technique of PID parameters.

With the development of swarm intelligence algorithm, researchers have used it for parameter tuning of PID controllers. Ouyang [4] used differential evolution (DE), genetic algorithm (GA) and PSO to optimize the gain of the position domain of PID controller to improve the contour tracking performance of robot manipulator. Wang [5] proposed a hydro-generator excitation controller based on PSO and fuzzy PID. The initial parameters of the controller were determined by the PSO and the
system was dynamically controlled by the FAPID. In addition, compared with the traditional PID and fuzzy PID control, the simulation results demonstrate that the fuzzy adaptive PID based on PSO algorithm had faster response speed and smaller overshoot. Jaafar [6] proposed a PID control method for a high nonlinear double pendulum bridge crane without payload motion feedback signal. Using the improved PSO algorithm, the optimal parameters of the PID controller were achieved according to the vertical distance oscillation and potential energy of the crane. Compared with the conventionally adopted PSO algorithm based on horizontal distance, this method achieved high performance for the less complicated controller.

In the evolutionary computational method, the PSO algorithm has robustness in solving continuous nonlinear optimization problems [7]. Compared with other methods, PSO method can produce high-quality solutions with less computational time and has stable convergence characteristics. However, the initial particle quality of the PSO algorithm is unstable, and it is easy to fall into local extremum during the iterative process [8]. In order to solve the shortcoming of PSO, a modified PSO algorithm based on chaos theory for PID tuning based on PMSM is proposed in this work. The tuning effect of the modified PSO algorithm on the parameters of the PI controller has been analyzed with the MATLAB/Simulink and the result indicates that this algorithm is effective for tuning the PI controller parameters of the PMSM servo system.

2. Mathematical model and methods

2.1. Mathematical Model of PMSM

The mathematical model of PMSM in rotor reference frame can be expressed as follows [9]:

\[ u_d = R_i d + L_d \frac{d}{dt} i_d - \omega L_i q \]
\[ u_q = R_i q + L_q \frac{d}{dt} i_q + \omega (L_i d + \psi) \]
\[ \frac{d}{dt} \omega = \frac{1}{J} (T_e - T_L - B \omega) \]

Where, \( u_d \) and \( u_q \) are the voltage in the q-axis and d-axis; \( i_d \) and \( i_q \) are the q-axis and d-axis current; \( R \) is stator resistance per phase; \( \psi \) is armature flux linkage due to rotor magnets; \( L_d \) and \( L_q \) are quadrature and direct axis stator self-inductance in rotor reference frames; \( T_e \) is the electromagnetic torque; \( T_L \) is the Load Torque; \( \omega \) is electrical rotor speed; \( \omega_m \) is mechanical rotor speed.

In the surface-mount PMSM, the stator self-inductance satisfies \( L_d = L_q = L_s \), and therefore the electromagnetic torque equation is:

\[ T_e = \frac{3}{2} p s q \psi \]

2.2. Chaos particle swarm optimization algorithm

In a D dimensional search space, the position of particle \( i \) is represented as \( x_i = (x_{i,1}, x_{i,2}, \ldots, x_{i,D})^T \), and the velocity of particle \( i \) is represented as \( v_i = (v_{i,1}, v_{i,2}, \ldots, v_{i,D})^T \). The best previous fitness and corresponding position of particle \( i \) is reflected as \( P_{best} = (P_{i,1}, P_{i,2}, \ldots, P_{i,D})^T \), respectively. Fitness and position of the best particle in the swarm is represented as \( g_{best} \) and \( P_{g} = (P_{g,1}, P_{g,2}, \ldots, P_{g,D})^T \). In the standard PSO, the search process can be express as [10]:

\[ v_{i}^{t} = w v_{i}^{t-1} + c_1 r_1 (P_{i}^{t-1} - x_{i}^{t-1}) + c_2 r_2 (P_{g}^{t-1} - x_{i}^{t-1}) \]
\[ x_{i}^{t} = x_{i}^{t-1} + v_{i}^{t} \]

Where, \( t \) is the iteration number; \( w \) is the inertia weight; \( r_1 \) and \( r_2 \) are uniformly distributed random numbers with range \([0,1] \); \( c_1 \) and \( c_2 \) are called cognitive and social acceleration coefficients respectively.
In the PSO algorithm, an evaluation function is selected to calculate the quality of particles’ position. The Integrated time and absolute error (ITAE) are used as the fitness function in the paper, which is defined as:

\[ F = \int_0^t [e(t)]^2 dt \]  

(7)

Logistic equation is a typical chaotic system, which is expressed as [11]:

\[ z(t+1) = \mu z(t) [1 - z(t)] \]  

(8)

Where, \( z \in [0,1] \) (\( z(0) \neq 0, 0.25, 0.5, 0.75, 1 \)), \( \mu \) is the control parameter and usually set to 4.

In the process of particle swarm initialization, chaos initialization can generate a large number of ordered chaotic particles and improve the quality of initial particles. Aiming at the problem that particles are easily trapped in local extremum in PSO algorithm, chaos disturbance is added in the search process to make particle jump out of local extremum and realize global optimization [12].

Chaos theory mainly includes chaos initialization and chaos disturbance. The process of CPSO is as follows:

1) Initialize the particle’s position and velocity with the chaos, and the initial groups are selected according to the principle of fitness-best selection.
2) Add a disturbance \( \Delta x \) to its existing position of each particle to produce a new particle. Calculate the fitness value of the new and old particle, and select the particle with better performance to form a new particle group.
3) Update the individual extremum and the global extremum to obtain corresponding particle.
4) Update the particle’s position and velocity according to expression (5) and (6).
5) Detect convergence condition. If the termination condition is satisfied, the iteration is terminated and the optimal solution is output. Otherwise, return (2).

2.3. Design of Speed Loop PI Controller Based on CPSO Algorithm

PMSM has speed control, torque control and position control. Speed control is the most common control method in servo systems. The performance of the system is determined by the speed loop. Therefore, CPSO algorithm is adopted to set the PI parameters of the speed loop in the paper. The process of tuning the speed loop PI parameters using CPSO algorithm is shown in Figure 1.

![Figure 1. The process of tuning the PI parameters using CPSO algorithm.](image-url)
3. Results and discussion

According to the PMSM vector control and the optimization principle of CPSO algorithm, the PMSM vector control schematic based on CPSO is implemented as follows.

The number of the chaotic particles and iterations are set at 100 and 30, and the reference speed is predetermined at 1000 r/min. According to the parameter optimization process of PI controller using CPSO, the fitness curve is shown in Figure 3(a), and the optimization curve of proportional coefficient $K_p$ and integral coefficient $K_i$ is shown in Figure 3(b) and (c), respectively.

![Figure 3](image)

**Figure 3.** The process of optimizing PI controller parameters using CPSO algorithm.

The results suggest that the system obtains the best parameter at the 10th iteration, before reaching the maximum number of iterations. As shown in Figure 3(b) and (c), the parameters set by CPSO algorithm are $K_p=0.21$ and $K_i=0.97$.

In order to verify the performance of the PI CPSO algorithm, the performance of the PMSM system using different tuning methods is compared. The traditional PI controller has been manually debugged with the parameters of $K_p = 0.14$ and $K_i = 7$. After optimization by PSO algorithm, the PI controller parameters are determined at $K_p=0.18$ and $K_i=3.95$. The parameters of the PI controller obtained by the CPSO algorithm are $K_p=0.21$ and $K_i=0.97$, respectively. The speed response curve corresponding to two different parameters of the PI controller in two cases is shown in Figure 4.
As shown in Figure 4, it is difficult to obtain satisfactory results by manual tuning, and the amount of overshoot is large. The PI controller parameters obtained by PSO algorithm have less overshoot than manual tuning, and the system based on PSO algorithm has faster response speed. Compared with PSO algorithm, CPSO algorithm has improved substantially in response to speed and overshoot indicating that the CPSO algorithm is effective for tuning the PI controller parameters of the PMSM servo system.

In order to quantify the performance difference of the three methods, the CPSO algorithm, the PSO algorithm and the manual tuning are compared from the following aspects, overshoot ($\sigma_p$), rise time ($t_r$), peak time ($t_p$), and integral of error and time (ITEA) (Table 1).

**Table 1. PMSM performance parameters.**

| Tuning method   | $\sigma_p$ (%) | $t_r$ (s)     | $t_p$ (s)     | ITEA   |
|-----------------|----------------|---------------|---------------|--------|
| Manual tuning   | 15.81          | $7.65 \times 10^{-3}$ | $13.54 \times 10^{-3}$ | 0.0584 |
| PSO             | 6.62           | $8.09 \times 10^{-3}$ | $11.89 \times 10^{-3}$ | 0.0334 |
| CPSO            | 6.10           | $8.11 \times 10^{-3}$ | $9.90 \times 10^{-3}$ | 0.0306 |

4. **Conclusion**

In this paper, the CPSO, a modified PSO algorithm based on chaos theory, has been developed to mitigate premature convergence problem of conventional PSO. The CPSO is applied to tuning the PI controller parameter for tuning the PMSM system. For validating CPSO, its performance is compared with conventional PSO and manual tuning method. The simulation results suggest that the CPSO algorithm can improve the tuning efficiency. In addition, the PMSM servo system based on CPSO algorithm has achieved better performance.

5. **Reference**

[1] Kommuri S K, Defoort M, Karimi H R and Veluvolu K C 2016 *IEEE Transactions on Industrial Electronics* **63** 7671-81
[2] Zhang X G, Hou B S and Mei Y 2017 *IEEE Transactions on Industrial Electronics* **32** 3818-34
[3] Shah P and Agashe S 2016 *Mechatronics* **38** 29-41
[4] Ouyang P and Pano V 2015 *Algorithms* **8** 697-711
[5] Wang Z, Zhang L and Peng Z 2017 Hunan Daxue Xuebao/Journal of Hunan University Natural Sciences 44 106-111
[6] Jaafar H I, Mohamed Z, Mohd S and Husain A R 2019 JVC/Journal of Vibration and Control 25 907-921
[7] Zhang J K 2009 Computer Engineering and Application 45 43-45
[8] Jordehi A R 2015 Applied Soft Computing 26 401-417
[9] Niu F, Wang BS, Babel AS and Li K 2016 IEEE Transactions on Power Electronics 31 1408-24
[10] Jordehi and Rezaee A 2015 Applied Soft Computing 26 401-417
[11] Tan D 2012 Lecture Notes in Electrical Engineering 144 469-476
[12] Liu J and Gao Y 2008 Journal of Computer Applications 28 322-325

Acknowledgements
The authors are grateful to the financial support from the National Natural Science Foundation of China (NO. U1813222), the National Natural Science Foundation of Guangdong (No.2016A030313177), Guangdong Frontier and Key Technological Innovation (No. 2017B090910013), the Science and Technology Innovation Commission of Shenzhen (No. JCYJ20170818153048647 and No. JCYJ20180507182239617).