

基于改进RAO算法的永磁同步电机2-DOF PI参数整定

唐高铭, 简献忠*

上海理工大学光电信息与计算机工程学院, 上海

收稿日期: 2024年1月23日; 录用日期: 2024年3月18日; 发布日期: 2024年3月26日

摘要

针对2-DOF PI控制器参数不易整定、RAO-1算法容易陷入局部解的缺点, 本文提出一种用于永磁同步电机参数整定的改进RAO (LILRAO)算法。在RAO算法中引入透镜成像的反向学习策略应用于种群更新, 避免了算法陷入局部最优; 引入Tent混沌映射进行种群初始化, 增强了算法全局搜索能力。在PMSM速度环中, 使用2-DOF PI控制器代替传统PI控制器, 提高系统跟随性和抗扰动特性; 测试函数实验表明LILRAO算法的有效性; Simulink仿真表明LILRAO算法与2-DOF PI控制器相结合可以有效提高PMSM控制系统的跟随性能和抗扰动性。

关键词

永磁同步电机, 二自由度控制, 参数整定, RAO算法

Optimum 2-DOF PI Controller Design in PMSM Using Improved RAO Algorithm

Gaoming Tang, Xianzhong Jian*

School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai

Received: Jan. 23rd, 2024; accepted: Mar. 18th, 2024; published: Mar. 26th, 2024

Abstract

This paper presents an improved RAO algorithm, named LILRAO algorithm, to address the challenges associated with difficult parameter tuning for the 2-DOF PI controller and the tendency of

*通讯作者。

the RAO-1 algorithm to converge to local solutions. The proposed algorithm focuses on parameter tuning for a permanent magnet synchronous motor. To prevent the algorithm from being trapped in local optima, a reverse learning strategy based on lens imaging is introduced and integrated into the population update process of the RAO algorithm. Moreover, the Tent chaotic map is employed for population initialization, thereby enhancing the algorithm's global search capability. In the PMSM speed loop, the traditional PI controller is replaced by the 2-DOF PI controller to enhance system tracking and disturbance rejection. Experimental results on test functions verify the effectiveness of the LILRAO algorithm, while Simulink simulations demonstrate that combining the LILRAO algorithm with the 2-DOF PI controller significantly improves the tracking performance and disturbance rejection of the PMSM control system.

Keywords

Permanent Magnet Synchronous Motor, Two Degrees of Freedom Control, Parameter Setting, RAO Algorithm

Copyright © 2024 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

永磁同步电机(PMSM)具有高效率、高功率密度、高转矩惯性比、低噪声、低维护等优点,已广泛应用于电动汽车、航空航天、机器人技术等领域。因此,它的控制研究受到越来越多的关注。目前市面上大多数永磁同步电机控制方法为闭环 PI 控制的矢量控制,与 PID 控制器相比,PI 控制器[1] [2] [3]不包含微分环节,不会引入高频噪声问题,且有较强的抗干扰能力。因此在 PMSM 中进行 PI 参数整定具有重要的理论和工程价值。

将智能控制融入到控制器的参数整定已成为主流控制方法。Visioli [4]在负载干扰的情况下采用模糊逻辑对 PID 控制器进行调优,提高了设定跟踪的性能。但是,对于模糊参数准则的选择仍然缺乏明确的解释。Kumar V [5]利用最小二乘法和人工神经网络相结合实现 PI 控制器参数自整定,实时更新控制器增益,有效减少了网络中需要调整的参数数量。然而,神经网络控制方法复杂,实现困难,对硬件要求高。随着群智理论的发展,出现了基于粒子群算法、蚁群算法和灰狼算法等方法的 PMSM 控制系统的参数整定和参数辨识[6] [7] [8]。文献[9]提出了一种基于混沌理论的改进粒子群算法(CPSO)用于 PMSM 的 PI 整定,以解决 PSO 算法初始粒子质量不稳定的问题。刘朝华[10]采用动态粒子群优化算法,利用高斯分布的动态反对学习策略,加强了全局搜索能力。传统的 PI 控制器为一自由度控制器,它的局限性在于只能整定一组参数满足一种期望性能,随着人们对电机控制系统的性能要求越来越高,一自由度控制无法满足人们的需求,提出了二自由度控制器对系统进行控制,利用一个自由度提升系统的鲁棒性,另一个自由度改善系统的跟随性。文献[11] [12] [13]提出了 2-DOF PI 控制器结构,用于永磁同步电机(PMSM)和无刷直流电机(BLDC)的控制,与传统 PI 控制器相比,所提出的控制器在设定响应、干扰抑制和鲁棒性方面取得了更好的性能。Hakan [14]使用二自由度 PI 控制器对永磁同步电动机进行磁场定向控制(FOC),以及使用 PSO 进行 2-DOF PI 控制器的参数整定,研究仅侧重于传统 PI 控制器与 2-DOF PI 控制器之间的比较,没有考虑可能提高 PMSM 性能的其他优化算法。尽管基于群智算法的参数整定方法取得了理想的结果,但是复杂的更新策略和大量的在线实时计算对系统硬件提出了挑战,将复杂控制算法移

植到嵌入式系统较为困难。然而, PSO 算法需要额外设置学习因子和粒子飞行速度, 灰狼算法(GWO)也需要设置探索距离变量。参数设置不当会导致计算量增加和寻优效果下降, 因此学者们开始关注无参数的优化方法和能够解决复杂问题的简单优化技术。Rao [15]提出了一种新型无参数非启发式群体优化算法。该算法根据种群与最优解、最差解和随机选择的解之间的相互作用, 在搜索空间中移动种群, 具有较强的局部开采能力和简单的更新策略公式, 并且具有良好的嵌入式实现优势。除了必要设置种群大小和迭代次数两个参数外, 该算法无需调节其他参数。研究发现, RAO 算法已成功应用于光伏电池参数辨识和寿命预测领域, 但尚未在 PMSM 的 PI 参数整定领域得到应用。

RAO 算法因初始种群存在随机性和更新环节的限制, 会导致算法全局开发能力不足和局部探索有随机性难以寻找到最优解, 影响 RAO 算法寻优精度。为了增强 PMSM 控制系统的稳定性, 本文提出了一种基于改进 RAO 算法, 算法将对 PMSM 的矢量控制(FOC) [11]中速度环的 2-DOF PI 控制器进行参数整定。首先, 为了改善随机初始种群全局探索不足导致的局部最优, 将 Tent 混沌映射序列引入种群的初始化。然后, 为了在局部进行更加细致的搜索, 在 RAO 算法更新环节加入透镜成像学习策略。LILRAO 算法是针对 PMSM 控制系统的 PI 参数整定做出的合理改进, 将 CPSO 算法和 LILRAO 算法应用于 PMSM 的 2-DOF PI 参数整定并比较。LILRAO 算法初始种群探索更加全面, 有效改善了 PMSM 伺服系统的控制性能。建立了 PMSM 的仿真模型和硬件实验平台, 比较了 CPSO 算法、GWO 算法、RAO 算法和 LILRAO 算法在 PMSM 的 2-DOF PI 参数方面的性能。结果表明, LILRAO 算法在初始种群中表现出更全面的探测能力, 有效地提高了 PMSM 伺服系统的控制性能。

2. PMSM 数学模型

对于 PMSM 的数学建模, 文献一般采用基于 d - q 轴的数学模型。利用 d - q 轴上的数学模型, 消除了时变电感方程, 使数学模型更加简单。因此, 更容易分析 PMSM 的动态行为。对于表贴式的 PMSM 来说, $L_d = L_q = L_s$, $i_d = 0$ 的控制方法最为简单, 忽略电机的磁滞损耗及定子铁芯饱和等问题, 在 d - q 坐标系下的 PMSM 数学方程如下:

$$\begin{cases} U_d = -\omega_e L_s i_q \\ U_q = R i_q + L_s \frac{di_q}{dt} + \omega_e \phi_f \\ Te = \frac{3}{2} pn \phi_f i_q = k_t i_q \\ J \frac{d\omega_m}{dt} = Te - TL - B\omega_m \end{cases} \quad (1)$$

式中: U_d , U_q 分别为定子电压的 d , q 轴分量; i_d , i_q 分别为定子电流的 d , q 轴分量; R 为定子电阻; ω_m 为实际机械角速度; ω_e 为电机转子电角速度; L_s 为定子电感; ϕ_f 为永磁体磁链; J 为机电系统转动惯量; B 为阻尼系数; Te 为电磁转矩; TL 为负载转矩; K_t 为电磁转矩常数; pn 为极对数。

3. 2-DOF PI 参数整定

3.1. RAO 算法理论

RAO 算法是一种无隐喻启发式的优化算法。多数智能算法如粒子群(PSO), 灰狼算法(GWO)。这类元启发式算法在更新机制上都采用了基于动物自然现象或行为隐喻的更新机制, 算法在提高寻优能力的同时, 无法避免引入一些额外参数, 而参数选择影响着求解的结果。RAO 算法根据最优解、最差解和随机选择的解之间的计算, 逐步靠近最优解, 远离最差解的方式进行更新。除了智能算法中常规的种群数

目与迭代次数两个参数需要设置, 无需其他参数设置。更新公式如下:

$$S'_{k,i} = S_{k,i} + rand * (S_{b,i} - S_{w,i}) \quad (2)$$

其中, $S'_{k,i}$ 是第 i 代的更新解, $S_{b,i}$, $S_{w,i}$ 是第 i 代种群候选解中的最好和最差值, $rand$ 为 $[0,1]$ 间的随机数。

3.2. LILRAO 算法设计

核心公式(2)目的是通过靠近最优解并避开最差解的寻优机制来寻找最优解的目的。此更新机制不需要设置额外参数, 但是寻优过程中会使得种群偏向局部最优。为了拓宽最优解的搜索领域, 改善算法的全局勘探能力, 本文对 RAO 算法进行改进。

首先引入 Tent 混沌映射代替正态分布产生的随机数 $rand$, 映射公式如下:

$$r_{n+1} = \begin{cases} \frac{r_n}{\alpha}, & r_n \in [0, \alpha] \\ \frac{(1-r_n)}{(1-\alpha)}, & r_n \in (\alpha, 1] \end{cases} \quad (3)$$

其中, n 为当前迭代次数, α 取值为 $(0, 1)$ 的随机数。

利用 Tent 混沌序列代替 RAO 算法中的随机数 $rand$, 利用 Tent 映射随机产生分布均匀的初始种群, 保证初始种群个体的随机性, 提升多样性, 从而提高算法的收敛速度。

在迭代过程中, 种群向最优解逼近, 会使得种群聚集在局部最优区域, 导致种群多样性降低。为了增强 RAO 算法的全局勘探能力, 本文在原有 RAO 算法的更新机制上增加基于透镜成像原理的反向学习策略, 将其运用在最优值的个体上产生新的个体与适应度值。

定义 1. 反向点: 假设 $X = (x_1, x_2, \dots, x_D)$ 为 D 维空间中的一个点, 且 $x_j \in [a_j, b_j]$ $j=1, 2, \dots, D$, 则 X 的反向点表示为 $X' = (x'_1, x'_2, \dots, x'_D)$, 且 $x'_j = a_j + b_j - x_j$ 。

定义 2. 基点: 若 D 维空间中存在若干个 o_1, o_2, \dots, o_m , 对于任意一点 $X = (x_1, x_2, \dots, x_D)$ 与其反向点 $X' = (x'_1, x'_2, \dots, x'_D)$ 到 o_i ($i=1, 2, \dots, m$) 的距离分别为 d_i 和 d'_i , 令 $k = d_i/d'_i$, 且 $k=1, 2, \dots, n$, 则 o_i 被称为 X 与 X' 在 $k=i$ 时的基点。

以一维空间为例, 假设有一高度为 h 的个体 P , 它在坐标轴上的投影为 x_{best} (x_{best} 为全局最优个体), 基点位置 o (本文取基点为 $[ub, lb]$ 的中点) 上放置焦距为 f 的透镜, 通过透镜成像过程可得到一个高度为 h' 的像 P' , 它在坐标轴上的投影为 x_{best}^* 。因此, 可得到全局最优个体 x^* 基于透镜成像原理的反向学习策略产生反向个体 x_{best}^* , 如图 1 所示。

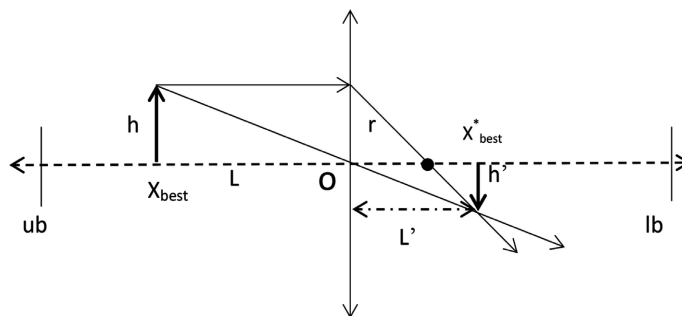


Figure 1. Schematic diagram of reverse learning strategy based on lens imaging
图 1. 基于透镜成像的反向学习策略示意图

在图 1 中, 最优个体 x_{best} 以 o 为基点得到其对应的反向点 x_{best}^* , 由透镜成像原理可得出:

$$\frac{\frac{u_b + l_b}{2} - X_{best}}{X_{best}^* - \frac{u_b + l_b}{2}} = \frac{L}{L'} = \frac{h}{h'} \quad (4)$$

令 $h/h' = k$, 称 k 为缩放因子, 对式(7)进行变换得到反向点 X_{best}^* 的计算公式:

$$X_{best}^* = \frac{u_b + l_b}{2} + \frac{u_b + l_b}{2k} - \frac{X_{best}}{k} \quad (5)$$

通过调整 k , 基于透镜成像策略得到的新个体是动态的, 从而进一步增强种群的多样性。以第 i 代的种群最优解 $S_{b,i}$ 为基准向量, 对最优解进行透镜成像学习策略, 产生反向解 $S_{b,i}^*$, 计算 $S_{b,i}^*$ 的适应度值, 若反向解的适应度值优于原最优个体的适应度值, 则将反向解个体替换原个体, 反之则保留原个体。

3.3. 测试函数实验

本小节选择 6 个测试函数对所提出的算法的寻优能力在 30 维上进行测试, 迭代次数为 1000, 每个测试函数重复运行 20 次以计算标准差。所用测试函数见表 1。其中, $f1, f2, f3$ 为单峰函数, $f4, f5, f6$ 为多峰函数。将结果与 PSO、GWO 和 RAO 算法进行比较。各算法比较结果分别见表 2。

Table 1. Test function

表 1. 测试函数

函数名	函数表达式	自变量范围	极值
Sphere	$f1 = \sum_{i=1}^n x_i^2$	[-100, 100]	0
Schwefel 2.22	$f2 = \sum_{i=1}^n x_i + \prod_{i=1}^n x_i $	[-10, 10]	0
Quadric	$f3 = \sum_{i=1}^n \left(\sum_{j=1}^i x_j \right)^2$	[-1.28, 1.28]	0
Ackley	$f4 = -20 \exp \left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2} \right) - \exp \left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i) \right) + 20 + e$	[-30, 30]	0
Rastrigin	$f5 = \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i) + 10)$	[-5.12, 5.12]	0
Griewank	$f6 = \sum_{i=1}^n \frac{x_i^2}{4000} - \prod_{i=1}^n \cos \left(\frac{x_i}{\sqrt{i}} \right) + 1$	[-600, 600]	0

Table 2. Test the experimental results of the function

表 2. 测试函数的实验结果

函数	CPSO			GWO			RAO-1			LILRAO		
	平均值	标准差	最优值	平均值	标准差	最优值	平均值	标准差	最优值	平均值	标准差	最优值
$f1$	1.86E+02	3.95E+02	6.63E+01	3.57E+02	3.24E+03	9.57E-60	3.21E-01	8.12E-02	1.89E-01	5.94E-07	1.88E-05	0.00E+00
$f2$	1.72E+24	5.42E+25	4.71E-05	3.82E+08	1.17E+10	1.18E-34	7.97E+10	5.63E+11	2.17E-07	1.18E-05	3.75E-04	0.00E+00
$f3$	1.72E+04	6.50E+03	1.58E+04	1.24E+03	9.15E+03	1.75E-18	3.60E+03	4.98E+03	7.43E+03	1.05E-06	3.31E-05	0.00E+00

续表

f_4	1.21E+01	1.22E+00	1.18E+01	2.25E+00	3.90E-01	2.22E-14	1.32E+00	5.51E-01	5.93E-02	5.70E-06	1.80E-04	8.88E-16
f_5	1.16E+02	4.24E+01	1.02E+02	1.99E+01	6.27E+01	0.00E+00	1.32E+01	3.82E+01	5.63E+00	3.40E-07	1.07E-05	0.00E+00
f_6	3.35E-01	7.88E-01	3.48E-11	3.40E+00	3.29E+01	0.00E+00	8.01E-02	6.95E-02	0.00E+00	1.13E-06	3.57E-05	0.00E+00

对比结果见表 2, 无论是单峰函数 $f_1 \sim f_3$, 还是多峰函数 $f_4 \sim f_6$, 本文所提出的算法寻优结果和标准差结果均优于 CPSO、GWO 以及 RAO 算法。对于单峰函数, 算法得到了理论最优解, 表明算法有较优的求解精度、求解速度和鲁棒性。在处理多峰函数中, 算法展现出了良好的全局探索能力, 且不易陷入局部最优。这表明了所提出的改进算法在寻优函数中的准确性和有效性。

3.4. 基于 LILRAO 算法的 2-DOF PI 控制器 PMSM 系统设计

矢量控制(FOC)的永磁同步电机的电流分为两个轴 d-q, 一个轴控制磁通, 另一个轴控制电流。因此, 转矩和磁通可以像直流电机一样相互独立地控制。矢量控制通常取 d 轴电流为零, 即 $i_d = 0$ 。基于 2-DOF PI 控制器的 PMSM 的 FOC 框图如图 2 所示。

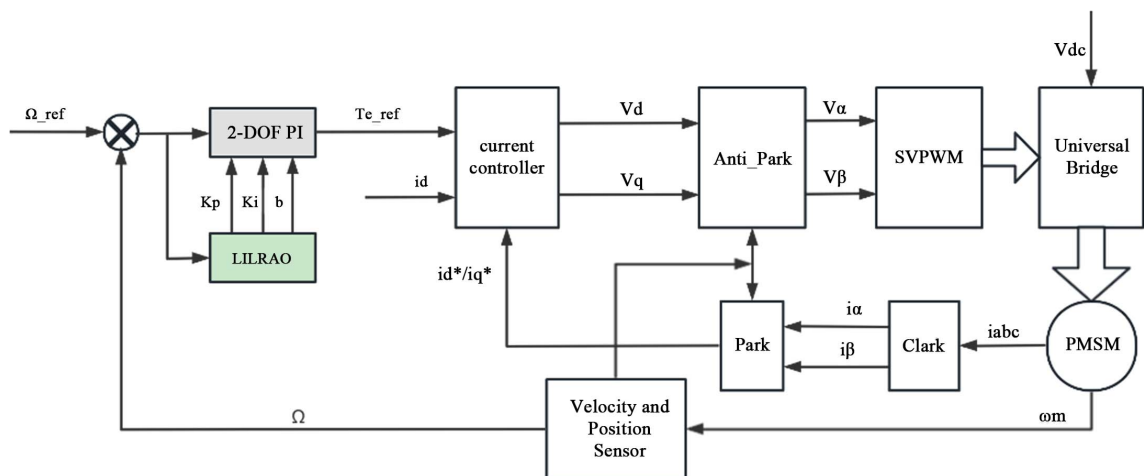


Figure 2. LILRAO algorithm for PMSM 2-DOF PI control system
图 2. LILRAO 算法的 PMSM 2-DOF PI 控制系统框

如方框图所示, 从 2-DOF PI 速度控制器的输出得到转矩基准 T_{e_ref} , 通过式 3 得到 i_q 基准电流。然后, 通过从电机测量的三相 i_a, i_b, i_c 电流, 用 Clark 和 Park 变换得到 i_d 和 i_q 轴电流。这些电流从参考 i_d 和 i_q 电流中减去, 并应用于 2 自由度 PI 电流控制器。 V_d 和 V_q 电压由控制器的输出获得。这些电压用于反 Clark 和 Park 变换, 以获得三相 V_a, V_b, V_c 参电压。在 PWM 模块中, 确定了电压源逆变器的开关信号。

采用 2-DOF PI 控制器代替传统 PI 控制器作为速度和电流控制器。这是因为 2-DOF PI 控制器比传统的 PI 控制器响应更快, 更适合于可变的操作条件[11] [12] [13] [14]。

2-DOF PI 控制器的特性表达式如式 6 所示。

$$G(s) = P(b \cdot r - y) + I \frac{1}{s}(r - y) \quad (6)$$

式中, $G(s)$ 为控制信号, P 为比例增益, I 为积分增益, b 为比例权重系数, r 为参考输入, y 为系统输出。

4. Simulink 仿真及分析

4.1. 评价指标

在 2-DOF PI 控制器的设计中, 使用速度参考与系统速度输出之间的积分时间加权绝对误差(ITAE)作为目标函数, 评价函数 ITAE 设置如下:

$$ITAE = \int_0^{\infty} t |e(t)| dt \quad (7)$$

采用 LILRAO 算法设计 2-DOF 控制器, 设定 P, I, b 参数范围, 将式(7)作为适应度函数, 由算法整定的参数, 使得适应度值最低。

4.2. 参数设置

仿真实验环境: AMD Ryzen 7 5800H CPU; 操作系统: Windows 11; 处理器速度: 3.20 GHz; 内存: 16.00GB; 编程环境: Matlab R2020b。

本文利用 LILRAO 算法在寻优空间内寻找一组参数 $\{P, I, b\}$, 使得系统满足设计要求, 具有良好的性能。在 Matlab/Simulink 平台下搭建了基于 LILRAO 算法的 2-DOF PI 控制的 PMSM 双闭环控制系统的仿真模型。PMSM 电机参数如表 3 所示。

Table 3. Electric machine parameter

表 3. 电机参数

参数	数值	参数	数值
N/(r/min)	1000	B/(N·m·s)	0.008
T/(N·m)	5	$\varphi_f/(W_b)$	0.204
P	4	R/(Ω)	12.8
J/(kg·m ²)	0.00017	$L_d(L_q)/(mH)$	302

考虑到系统控制中需要整定参数, 将变量数取为 N; 种群数量设置为 10; 迭代次数为 30; 缩放因子为 1000。基于 LILRAO 算法的 PMSM 2-DOF PI 控制器参数整定步骤如下:

Step 1: 初始化种群数量 pop、变量数 var、迭代次数 T、缩放因子 K、上界 ub、下界 lb。

Step 2: 应用式(2) (3) (7)对种群进行初始化, 计算适应度值。

Step 3: 根据适应度值, 选择三组较优粒子进行式(5)的反向学习更新策略; 其他粒子选择式(2) (3)进行更新。

Step 4: 计算新粒子适应度值, 并与原粒子比较, 结果更好则替换。

Step 5: 重复 Step 3 至 Step 4 直至满足迭代条件。

Step 6: 迭代结束, 得到最优的 P, I, b , 参数并输出。

首先对比 PI 控制器与 2-DOF PI 控制器的控制效果, 设置转速在 0~1 s 时为 1000 r/min, 0.5 s 时突加额定负载 5 N·m, 在 1 s 时转速变为 500 r/min。图 3 为 PI 控制器与 2-DOF PI 控制器的电机转速的响应曲线, 图 4 为 PI 控制器与 2-DOF PI 控制器的转速误差曲线。

根据图 3 与图 4 结果可知, 2-DOF PI 控制器对比 PI 控制器明显改善系统的跟踪性能, 在 0.5 s 时突加负载, 2-DOF PI 控制系统能很快适应负载力, 有更快的动态负载特性。

为了评估系统的动态响应和稳态行为, 将电机转速设置为额定转速 1000 r/min, 在 0.5 s 时突然施加

额定负载 5 N。图 5 为电机转速的响应曲线, 由转速响应曲线得到的仿真性能比较结果见表 4。在仿真实验中, 对四种控制策略进行了比较分析。

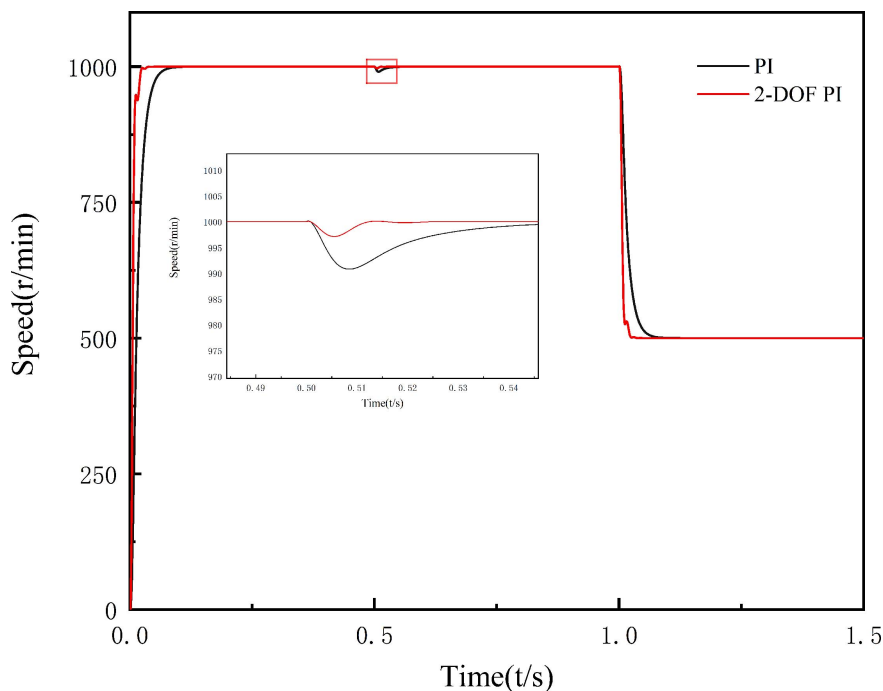


Figure 3. PI controller and 2-DOF PI controller motor speed response curve
图 3. PI 控制器与 2-DOF PI 控制器的电机转速的响应曲线

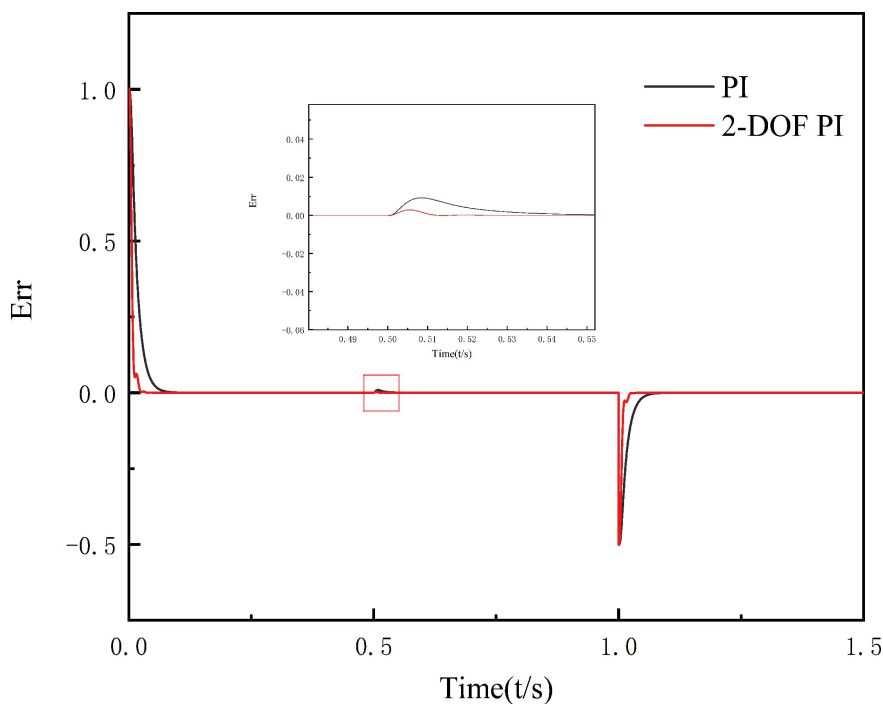


Figure 4. PI controller and 2-DOF PI controller motor speed error curve
图 4. PI 控制器与 2-DOF PI 控制器的转速误差曲线

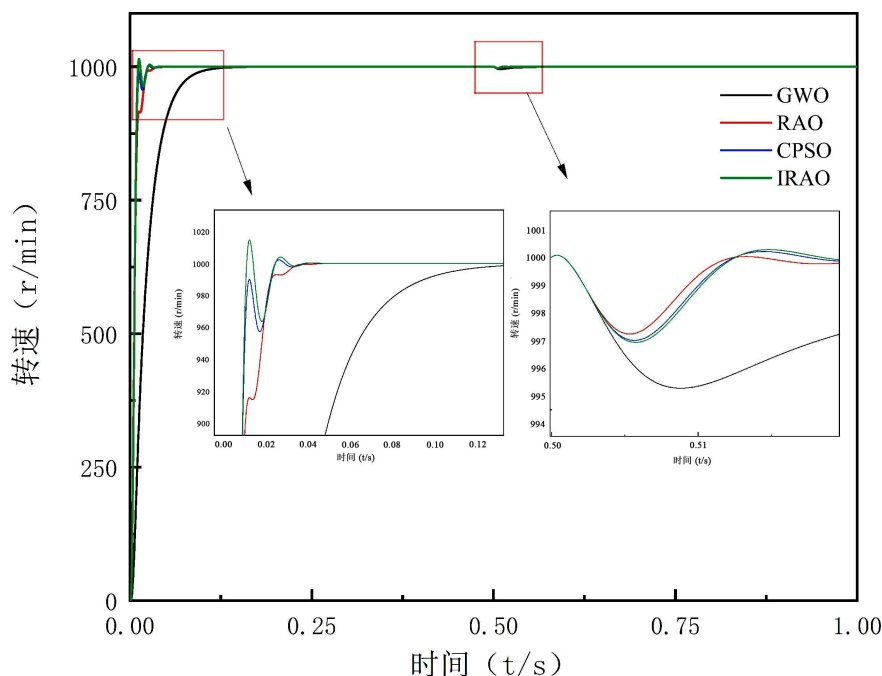


Figure 5. Different algorithms of 2-DOF PI control the response curve of motor speed
图 5. 不同算法的 2-DOF PI 控制电机转速的响应曲线

Table 4. Comparison results between controller parameters and simulation performance indexes
表 4. 控制器参数与仿真性能指标对比结果

整定方法	Kp	Ki	b	Tr/s	Mp/%	Ts/s	Ess/%
GWO	3.0	0.01	0.04	0.0129	3.85	0.031	4.11
RAO	2.0	0.08	0.02	0.0112	2.11	0.030	4.01
CPSO	2.5	0.015	0.02	0.0152	3.56	0.0348	4.23
LILRAO	3.2	0.02	0.02	0.0112	1.72	0.025	3.66

仿真实验使用 4 组控制策略进行对比：通过图 5 和表 4 的结果可以看出本文提出的 LILRAO 算法对于 2-DOF PI 控制器有较好的寻优效果；无论是抗扰动性能和目标转速跟踪性能都优于其他三组参数得到的仿真结果。

5. 硬件实验

硬件实验平台如图 6 所示，由一台 0.2 kW 的永磁同步电机、电机驱动器、微控制器、主机、隔离变压器和示波器组成。本文采用 TI 公司的浮点式 DSPTMS320F28335 控制器，实验中使用的伺服电机参数如表 3 所示。电机矢量控制算法的设计是在代码编写器工作室软件中进行的。该系统控制程序由一个主程序和一个循环定时器中断程序组成。中断功能在定时器周期内执行 A/D 采样、位置和速度估计、坐标变换、速度和当前 PI 控制以及 SVPWM 调制等任务。利用微控制器与主机之间的实时通信，获得了电机在运行过程中的转速、转子位置和电流数据。

实验验证：在空载条件下，对不同算法的电机参数调整进行了实验测试。本实验的电机参数与仿真设置参数相同，电机 2-DOF PI 控制器的参数与表 4 一致。

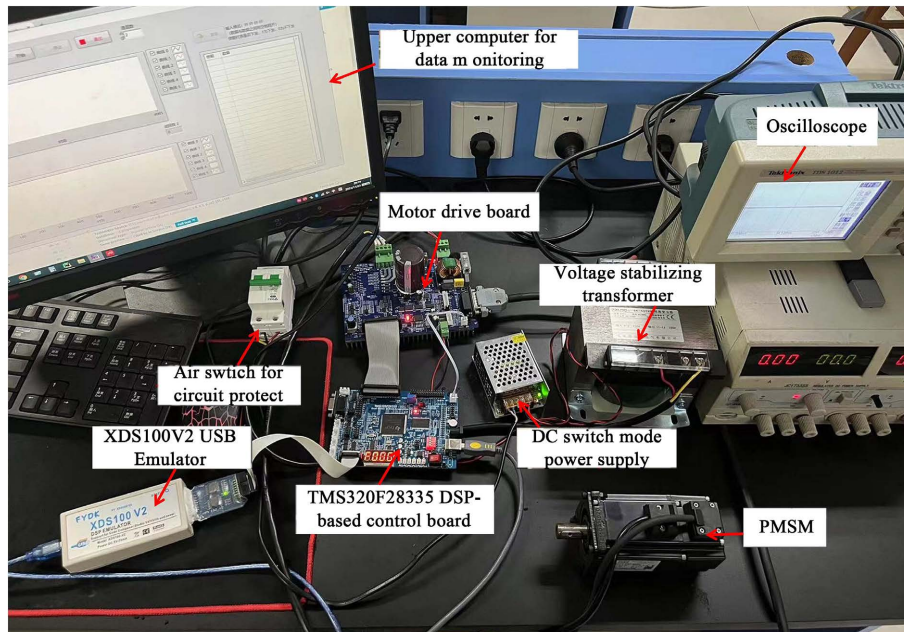


Figure 6. Experimental platform
图 6. 实验平台

Table 5. Comparison results between controller parameters and simulation performance indexes
表 5. 控制器参数与硬件实验性能指标对比结果

整定方法	Kp	Ki	b	Tr/s	Mp/%	Ts/s	Ess/%
GWO	3.0	0.01	0.04	0.0113	2.2	0.024	5.51
RAO	2.0	0.08	0.02	0.0117	1.6	0.033	5.73
CPSO	2.5	0.015	0.02	0.0124	1.5	0.030	5.69
LILRAO	3.2	0.02	0.02	0.0116	1.3	0.014	4.64

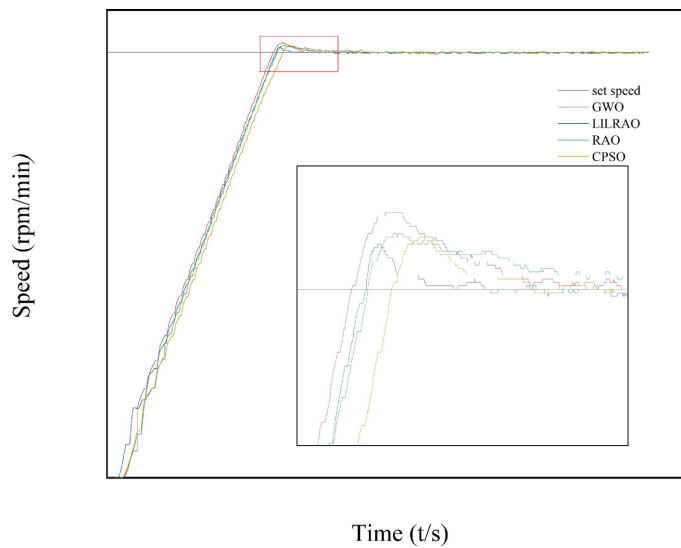


Figure 7. Different algorithms of 2-DOF PI control the response curve of motor speed in the hardware platform
图 7. 不同算法的 2-DOF PI 控制电机转速的响应曲线

表 5 与图 7 为硬件实验不同算法的 2-DOF PI 控制电机转速的性能指标与速度响应曲线。

与仿真结果相比, 在图 7 中, 相同工况下电机运行状态不同, 主要是由于传感器精度、机械噪声和外部环境对电机的干扰。在表 5 中, 在用于 2-DOF PI 参数调整的不同算法中, LILRAO 算法的 ITAE 值最低, 为 4.64%, 显示出相对稳定的操作状态。相比之下, RAO 算法在稳定运行过程中存在振动, 参数调优结果属于局部最优。GWO 算法的问题是本实验的过调过大, 这与仿真结果一致。对于 PSO 算法, 上升时间相对较大, ITAE 值与 GWO 算法相似。

图 8 和图 9 为 LILRAO 算法在空载模式下的电流和转子位置数据图。在 0.014 s 时, 电机完成启动, 由于运行过程中的机械噪声和传感器精度问题, 出现了一些误差值。实验表明, LILRAO 算法对参数调优是可行的。

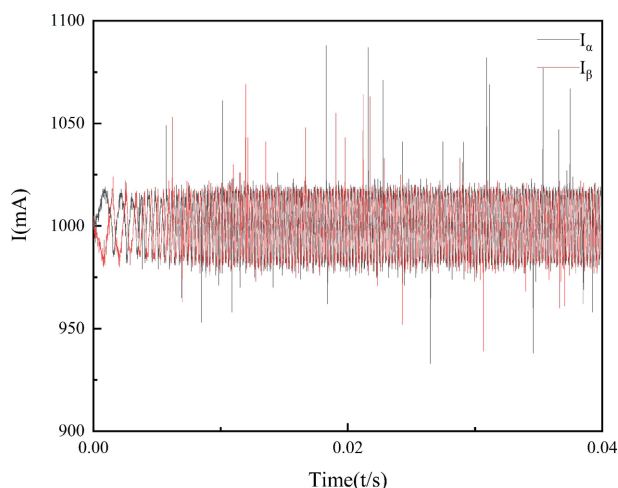


Figure 8. Current diagram of LILRAO algorithm in no-load mode

图 8. 空载模式下 LILRAO 算法的电流图

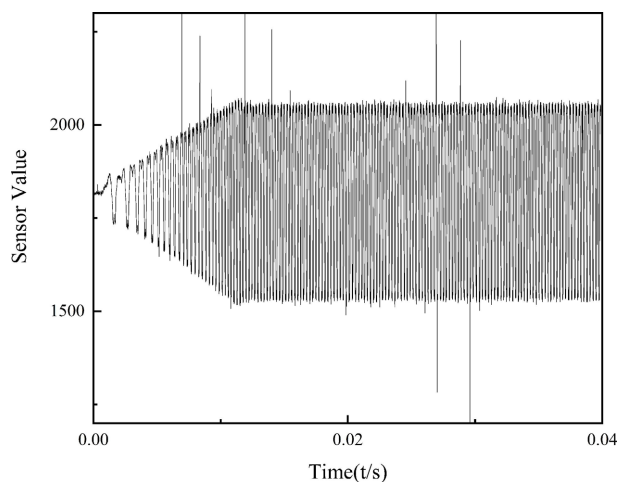


Figure 9. Sensor value diagram of LILRAO algorithm in no-load mode

图 9. 空载模式下 LILRAO 算法的传感器值图

6. 结论

本文介绍了基于所提出的 LILRAO 算法的永磁同步电机(PMSM)的 2-DOF PI 控制器的参数调优方

法。算法初始化过程结合了 Tent 混沌映射, 在算法迭代过程中, 采用了基于透镜成像的学习策略。这些方法在保持了初始种群的多样性的同时, 提高了探索和在整个优化过程中跳出局部最优的能力。该算法提高了控制器参数搜索的速度和精度。通过对 PMSM 速度环控制的仿真实验, 该方法在上升时间、超调、沉降时间和 ITAE 方面具有良好的性能, 带有机噪声的硬件实验也满足预期要求。基于 LILRAO 算法的 2-DOF PI 控制方法可以进一步应用于未来的嵌入式开发过程中。

基金项目

本文由国家自然科学基金资助项目 11774017。

参考文献

- [1] Chai, S., Wang, L. and Rogers, E. (2013) A Cascade MPC Control Structure for a PMSM with Speed Ripple Minimization. *IEEE Transactions on Industrial Electronics*, **60**, 2978-2987. <https://doi.org/10.1109/TIE.2012.2201432>
- [2] Xu, Y., Parspour, N. and Vollmer, U. (2014) Torque Ripple Minimization Using online Estimation of the Stator Resistances with Consideration of Magnetic Saturation. *IEEE Transactions on Industrial Electronics*, **61**, 5105-5114. <https://doi.org/10.1109/TIE.2013.2279378>
- [3] 邵雪琳, 王亚君, 刘海辉, 等. 智能电机控制系统[J]. 电子世界, 2016(22): 89-90.
- [4] Visioli, A. (2001) Tuning of PID Controllers with Fuzzy Logic. *IEE Proceedings-Control Theory Applications*, **148**, 1-8. <https://doi.org/10.1049/ip-cta:20010232>
- [5] Kumar, V., Gaur, P. and Mittal, A.P. (2014) ANN Based Self Tuned PID Like Adaptive Controller Design for High Performance PMSM Position Control. *Expert Systems with Applications*, **41**, 7995-8002. <https://doi.org/10.1016/j.eswa.2014.06.040>
- [6] 张铸, 姜金美, 张小平. 改进灰狼优化算法的永磁同步电机多参数辨识[J]. 电机与控制学报, 2022, 26(10): 119-129. <https://doi.org/10.15938/j.emc.2022.10.014>
- [7] 石川东. 改进粒子群算法的永磁同步电机参数辨识[D]: [硕士学位论文]. 株洲: 湖南工业大学, 2022. <https://doi.org/10.27730/d.cnki.gnhgy.2022.000442>
- [8] 蔡琦盼. 基于改进混沌粒子群算法和灰狼算法的永磁同步电机参数辨识[D]: [硕士学位论文]. 赣州: 江西理工大学, 2022. <https://doi.org/10.27176/d.cnki.gnfyc.2022.000265>
- [9] Zuo, R., Xiong, X., et al. (2019) Design of PI Controller for PMSM Using Chaos Particle Swarm Optimization Algorithm. *IOP Conference Series: Materials Science and Engineering*, **717**, Article 012021. <https://doi.org/10.1088/1757-899X/717/1/012021>
- [10] Liu, Z., Wei, H., Zhong, Q., et al. (2017) Parameter Estimation for VSI-Fed PMSM Based on a Dynamic PSO with Learning Strategies. *IEEE Transactions on Power Electronics*, **32**, 3154-3165.
- [11] 李明, 程启明, 陈根, 等. 永磁同步伺服电机二自由度控制[J]. 电机与控制应用, 2014, 41(10): 3-7+27.
- [12] Stfnean, A.-I., Preitl, S., et al. (2012) 2-DOF PI(D) Takagi-Sugeno and Sliding Mode Controllers for BLDC Drives. *2012 15th International Power Electronics and Motion Control Conference*, Novi Sad, 4-6 September 2012, 1-6.
- [13] 蔡豪, 汪亮亮. 二自由度 PID 在 PMSM 电流环中的应用[J]. 机电工程技术, 2017, 46(7): 26-28+143.
- [14] Celik, H. and Yigit, T. (2018) Field-Oriented Control of the PMSM with 2-DOF PI Controller Tuned by Using PSO. *2018 International Conference on Artificial Intelligence and Data Processing*, Malatya, 28-30 September 2018, 1-4.
- [15] Rao, R.V. (2020) Rao Algorithms: Three Metaphor-Less Simple Algorithms for Solving Optimization Problems. *International Journal of Industrial Engineering Computations*, **11**, 107-130. <https://doi.org/10.5267/j.ijec.2019.6.002>