

文章编号:1671-5896(2026)04-0816-08

基于个性化策略改进捕鱼算法的光伏电站 MPPT 优化

李 鹏

(大庆油田电力能源有限公司 总部机关, 黑龙江 大庆 163414)

摘要: 针对捕鱼优化算法(CFOA: Catch Fish Optimization Algorithm)在探索与开发2个更新阶段容易陷入局部最优且收敛速度较低的问题, 基于个性化捕鱼策略提出改进型捕鱼优化算法(ICFOA: Improved Catch Fish Optimization Algorithm)。首先, 在探索阶段引入自适应高斯扰动, 在提升全局搜索能力与效率的同时避免陷入局部最优。其次, 基于个性化捕鱼策略, 通过随机选择“徒手捕鱼”因子或“使用渔网”因子更新渔民位置, 以加速算法收敛。最后, 利用CEC2020测试套件进行对比实验, 评估ICFOA与其他优秀元启发式算法的性能差异, 并使用Wilcoxon秩和检验验证统计结果的有效性。其在光伏电站最大功率点跟踪应用结果表明, ICFOA能有效降低功率追踪差值。证明了解决实际问题的有效性, 相较原始CFOA具有更强的竞争力。

关键词: 捕鱼优化算法; 元启发式算法; 群体智能; 个性化捕鱼策略

中图分类号: TP18

文献标志码: A

Improved Catch Fish Optimization Algorithm with Personalized Strategy for Photovoltaic MPPT

LI Peng

(Headquarters, Daqing Oil Field Power Energy Company Limited, Daqing 163414, China)

Abstract: CFOA(Catch Fish Optimization Algorithm) usually includes two update stages, the exploration stage and the exploitation stage. However, this algorithm is still prone to getting stuck in local optima and has a relatively low convergence rate. To address these issues, the ICFOA (Improved Catch Fish Optimization Algorithm) is proposed based on personalized fishing strategies. Firstly, an adaptive Gaussian perturbation is introduced in the exploration stage to enhance the global search ability and efficiency while avoiding local optima. Secondly, based on personalized fishing strategies, the positions of fishermen are updated by randomly selecting either the “bare-handed fishing” factor or the “using a fishing net” factor to accelerate the convergence. The CEC2020 test suite is used to conduct comparative experiments to evaluate the performance differences between ICFOA and other excellent meta-heuristic algorithms, and the Wilcoxon rank sum test is used to verify the validity of the statistical results. Finally, the maximum power point tracking in photovoltaic power plants, ICFOA effectively reduces the power tracking deviation, demonstrating its effectiveness in solving practical problems. Experimental results indicate that ICFOA possesses greater competitiveness compared to the original CFOA.

Key words: catch fish optimization algorithm; meta-heuristic algorithms; swarm intelligence; personalized fishing strategy

收稿日期: 2025-12-13

基金项目: 黑龙江省重点研发计划基金资助项目(2024ZXJ01A04)

作者简介: 李鹏(1973—), 男, 山东日照人, 大庆油田电力能源有限公司高级工程师, 主要从事智能配电网、新能源风光场站运行管理研究, (Tel)86-18795741332 (E-mail)13359812455@126.com。

0 引言

群智优化算法通过模仿自然界的集体行为,运用简单规则和局部信息交换解决复杂优化问题^[1-4]。典型代表包括小龙虾优化算法(COA: Crayfish Optimization Algorithm)^[5-6]、鲸鱼优化算法(WOA: Whale Optimization Algorithm)^[7-8]和灰狼优化算法(GWO: Grey Wolf Optimizer)^[9-10]。其中 COA 擅长探索但收敛缓慢;WOA 在探索与开发间取得良好平衡,但易陷入局部最优;GWO 收敛能力强,但需精细的参数调优。Jia 等^[11]于 2024 年提出一种受人类行为启发的全新优化算法——捕鱼优化算法(CFOA: Catch Fish Optimization Algorithm)。该算法包含基于不同捕鱼方式的更新规则。但其也面临许多优化算法存在的问题^[12-16],即原始 CFOA 在特定优化任务中难以完全避免收敛效率低的局限性,且容易陷入局部最优解。对 CFOA 进行优化升级不仅能提升其算法效率,还能拓宽其应用范围。因此,笔者基于个性化捕鱼策略(PFS: Personalized Fishing Strategy)提出改进型 CFOA 算法(ICFOA: Improved Catch Fish Optimization Algorithm)。该策略能显著提升 CFOA 在复杂优化问题中的求解性能。通过个性化捕鱼策略更新渔民位置,不仅使算法在搜索解空间时更精细、全面,还增强了算法跳出局部最优、寻找全局最优解的能力。为验证 ICFOA 的改进效果,笔者采用 10 个常用基准函数,选取原始 CFOA 及 5 种代表性元启发式算法进行对比实验,结果充分验证了 ICFOA 的有效性与优势。

1 捕鱼优化算法

CFOA 包含初始化、探索和开发 3 个阶段。

1.1 初始化阶段

设 F 是 N 个搜索代理在 d 维空间中的位置信息的综合矩阵,其初始化公式如下:

$$F = \begin{bmatrix} F_{1,1} & F_{1,2} & \cdots & F_{1,n} \\ F_{2,1} & F_{2,2} & \cdots & F_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ F_{n,1} & F_{n,2} & \cdots & F_{n,n} \end{bmatrix}_{N \times d}, \quad (1)$$

$$F_{i,j} = l_j + (u_j - l_j)r, \quad (2)$$

其中 $F_{i,j}$ 是第 i 个渔民在第 j 维度的位置, u_j 和 l_j 分别是第 j 维度的上下界, r 是 $(0,1)$ 区间内的随机数。

基于每位渔民的当前位置信息,使用适应度值评估函数 f 计算其适应度值,并获得适应度值矩阵如下:

$$f = f(F) = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_N \end{bmatrix}, \quad (3)$$

其中 f_1 是第 1 位渔民的适应度值, f_2 是第 2 位渔民的适应度值,依此类推。采用 0.5 作为阈值,在迭代过程中均衡分配开发与探索的比例。即个体在前半段(当开发效率/最大开发效率 < 0.5 时)进行全局探索,在后半段(当开发效率/最大开发效率 ≥ 0.5 时)转为开发策略。

1.2 个体和群体捕鱼(探索阶段)

渔民在探索初期主要依靠独立搜索,辅以群体围捕。随着探索深入,环境优势逐渐从鱼类转向渔民,并且持续捕捞将导致鱼群数量和捕获率下降。渔民将从独立探索转向主要依靠集体围捕,个人优势作为辅助手段。该模式的转变通过捕获率参数 δ 进行模拟:

$$\delta = \left(1 - \frac{3 \times m}{2 \times M}\right)^{3 \times m / (2 \times M)}, \quad (4)$$

其中 m 是当前评估次数, M 是最大评估次数。

1.2.1 个体捕捞

当 $m/M < 0.5$ 时, 渔民搅动水面使鱼群浮起, 从而确定鱼群位置并调整搜索方向。公式更新如下:

$$E_{xp} = \frac{f_i - f_p}{f_{\max} - f_{\min}}, \quad (5)$$

$$R = D_{is} \sqrt{|E_{xp}|} \left(1 - \frac{m}{M}\right), \quad (6)$$

$$F_{i,j}^{T+1} = F_{i,j}^T + (F_{p,j}^T - F_{i,j}^T)E_{xp} + rsR, \quad (7)$$

其中 E_{xp} 是渔民以任意第 p 位 ($p = 1, 2, \dots, N, p \neq i$) 渔民为参照对象获得的经验分析值, 取值范围为 $(-1, 1)$ 。 $f_{\max}$ 和 $f_{\min}$ 分别是第 T 次完整位置更新后的最差与最佳适应度值。 T 是渔民位置迭代次数。 $F_{i,j}^T$ 与 $F_{i,j}^{T+1}$ 分别是第 i 位渔民在第 j 维度上经历第 T 次与第 $(T+1)$ 次迭代后的位置。 D_{is} 是第 i 个体与参照对象间的欧几里得距离。 s 是维度为 d 的随机单位向量。

1.2.2 群体捕捞

当 $m/M \geq 0.5$ 时, 渔民使用渔网扩大捕捞能力, 并与其他渔民协作。渔民随机组成 3~4 人的小组进行合作, 围堵目标对象。通过发挥各自独特的机动性, 更精准地探索该区域。模型如下:

$$C_c = e(F_c^T), \quad (8)$$

$$F_{c,i,j}^{T+l} = F_{c,i,j}^T + r_2(C_c - F_{c,i,j}^T) + \left(1 - \frac{2 \times m}{M}\right)^2 r_3. \quad (9)$$

其中 c 组是由 3~4 名成员组成, 且其位置尚未更新。 C_c 是 c 组实施包围战术的目标点。 $F_{c,i,j}^{T+l}$ 与 $F_{c,i,j}^T$ 分别是 c 组第 i 名渔民在第 j 维度上经历第 $(T+1)$ 次与第 T 次更新后的位置。 r_2 是渔民向中心靠近的速度, 因人而异且取值范围为 $(0, 1)$ 。 r_3 是移动偏移量, 数值范围为 $(-1, 1)$, 其值会随 m 增加而逐渐减小。

1.3 集体捕获(开发阶段)

所有渔民都采用统一策略进行搜索, 有目的地将隐藏的鱼群引至同一位置及其周边。捕捞过程中渔民的位置更新如下:

$$\sigma = \sqrt{\frac{2\left(1 - \frac{m}{M}\right)}{\left(\left(1 - \frac{m}{M}\right)^2 + 1\right)}}, \quad (10)$$

$$F_i^{T+1} = G_{\text{best}} + D\left(0, \frac{r_4 \sigma |e(F) - G_{\text{best}}|}{3}\right), \quad (11)$$

其中 D 是高斯分布函数, 其总体均值 μ 为 0, 总体方差 σ 随评估次数增加而增大, 从 1 趋近于 0。 F_i^{T+1} 是第 i 位渔民在第 $(T+1)$ 次更新后的位置。均值矩阵 (F) 是渔民中心各维度的均值矩阵。 G_{best} 是全局最优位置。

2 改进的捕鱼算法

2.1 自适应高斯扰动

在算法种群初始化时加入高斯扰动以改善其分布。自适应高斯扰动 (AGP: Adaptive Gaussian Perturbation) 通过

$$\sigma_p = \left(1 - \frac{m}{M}\right) s_{\text{std}}(F_i^T) \exp\left(-\frac{m}{M/10}\right), \quad (12)$$

$$F_i^{T+1} = F_i^T + \mathcal{N}(0, \sigma_p^2) \quad (13)$$

动态调整扰动强度, 使优化算法在不同迭代阶段平衡探索与开发。其中 σ_p 是当前迭代的扰动强度, s_{std} 是当前解的标准差, $\mathcal{N}(0, \sigma_p^2)$ 是均值为 0、方差为 σ_p^2 的正态分布随机变量。

2.2 个性化捕捞策略(PFS)

PFS(Presonalized Fishing Strategy) 是一种广受欢迎且高效的策略, 有助于提升算法在搜索空间中的优化

能力。并且 PFS 能增强算法的开发能力及提供更快收敛速度。其原理是基于原始动作步生成随机动作 α 和 β ，根据不同移动路径更新位置，并依据捕获参数 δ 进行调整，使整个捕捞过程更具关联性：

$$F_i^{T+1} = F_i^T + s_{\text{tep}}k, \tag{14}$$

$$k = 0.5 + 0.5r, \tag{15}$$

其中 s_{tep} 是行动选择， k 是渔民的行动技能熟练度，其取值范围为 $[0.1, 1]$ 。个性化捕鱼策略包含“徒手捕捞”和“使用工具”两个因素，用于提升算法的全局搜索能力。则更新后有

$$s_{\text{tep}} = \begin{cases} \alpha = 2\left(1 - \frac{m}{M}\right)(0.4\delta)r, & s_{\text{tep}} > 0.5, \\ \beta = 1 - \frac{m}{M}, & \text{其他。} \end{cases} \tag{16}$$

2.3 ICFOA 的详细信息

虽然 CFOA 算法易于实现且适用于广泛的优化任务，但在解决复杂优化问题时，其探索与开发能力不足，难以在有限迭代次数内收敛。为此，笔者引入 PFS 算法对 CFOA 进行优化以实现全局优化。PFS 用于增强传统 CFOA 的开发能力并加速其收敛速度。ICFOA 的流程如图 1 所示。

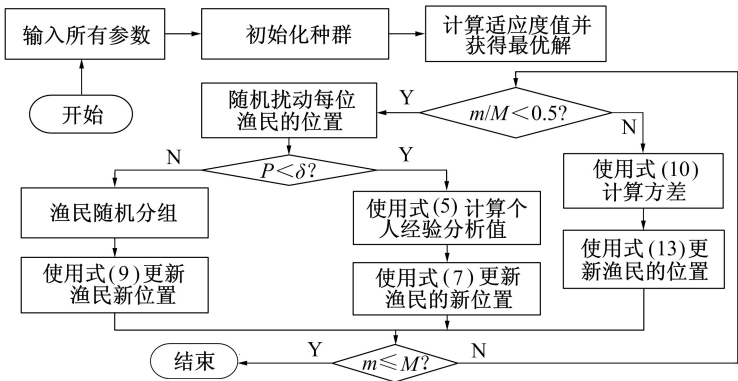


图 1 ICFOA 算法的流程

Fig. 1 Flowchart of the ICFOA algorithm

2.4 ICFOA 的计算复杂度

初始化和位置更新是 ICFOA 算法的核心部分。初始化的计算复杂度为 $O(N \times D)$ ，其中 N 是种群规模， D 是维度大小。徒手与使用渔网捕鱼存在不确定性，其最大计算复杂度分别为 $O(T \times N \times D)$ ， T 是最大评估次数。PFS 与 AGP 方法的计算复杂度均为 $O(T \times N \times D)$ 。综上，ICFOA 的总计算复杂度为 $O((4 \times T + 1) \times N \times D)$ 。

3 全局优化的实验结果

笔者采用 CEC2020 中的 10 个基准函数，评估所提 ICFOA 算法的优化性能。首先给出这 10 个基准测试函数的定义，其次详细描述实验设置及对比组，其中包含其他知名的优化算法。

3.1 10 个基准函数的定义

基准函数是评估算法性能的重要工具。10 个基准函数的定义如表 1 所示。

表 1 10 个基准函数的定义

Tab. 1 Definitions of 10 basis functions

函数	属性	维度	范围	最小值	函数	属性	维度	范围	最小值
F_1	单峰函数			100	F_6	混合函数	100	$[-100, 100]D$	1 600
F_2				1 100	F_7				2 100
F_3				700	F_8				2 000
F_4				1 900	F_9	复合函数			2 400
F_5	基础函数			1 700	F_{10}				2 500

笔者将选取的 10 个典型基准函数分为两类: 1) 单峰函数($F_1 \sim F_4$); 2) 多峰函数($F_5 \sim F_{10}$)。这些函数定义如表 1 所示, 其中 D 是维度。由于单峰函数仅存在唯一全局最优解, 可用于评估多代理系统的开发能力; 而多峰函数则存在多个局部最优解与唯一全局最优解, 可用于评估多代理系统探索局部最优解并跳出局部的能力。

3.2 实验设置

笔者使用上述函数测试 ICFOA 的性能。为使实验更具代表性, 将改进算法与基础 CFOA 及 5 种主流元启发式算法进行对比, 包括 ROA (Remora Optimization Algorithm)、AOA (Arithmetic Optimization Algorithm)、SFO (Sailfish Optimization)、SHO (Sea-Horse Optimization) 和 SCA (Sine Cosine Algorithm)。为确保测试公平性, 统一设定最大评估次数 $T=100\ 000$ 、种群规模 $N=30$ 及维度数 $D=10$ 。所有测试均独立运行 30 次。检验函数的统计量结果如表 2 所示。

表 2 CEC2020 中 10 个检验函数的统计量结果

Tab. 2 Statistics table for 10 verification functions in CEC2020

函数	指标	ICFOA	CFOA	ROA	AOA	SFO	SHO	SCA
F_1	最值	123.916 3	1 303.684	2 435.264	2 484.316	2 898.991	27 552.632	17 016.33
	均值	4 591.728	2 122.393	26 245.211	26 812.311	28 990.512	28 799.225	18 504.45
	标准差	4 137.802	6 844.681	1 044.941	908.347 48	192 406.9	4 948.401	13 440 911
F_2	最值	12 244.11	16 605.712	29 808.776	28 990.186	35 461.621	4 698.949	31 568.08
	均值	18 161.78	23 564.866	31 381.508	30 430.724	35 488.166	36 163.210	32 488.08
	标准差	3 545.047	4 192.739	114 655.13	9 077.919	24 045.264	103 551.20	5 910.424
F_3	最值	894.505 6	1 683.533	3 978.813	3 806.349	4 217.387	4 066.605	3 489.622
	均值	927.346 4	2 120.641	4 011.883	3 909.740	4 221.836	4 204.438	3 765.087
	标准差	40.844 72	270.976 8	56.032 661	53.705 448	30.762 791	75.210 840	180.655 9
F_4	最值	1 911.571	1 972.715	1 926.112 8	1 913.360	1 942.664	1 926.352	5 871.613
	均值	1 903.184	2 003.719	1 938.664	1 911.360	1 908.059	1 933.524	33 306.19
	标准差	1.974 573	20.759 07	48.005 463	7.623 121	5.236 162	10.326 585	41 611.33
F_5	最值	771 627.8	2 622.821	648.036 49	897.453 83	1 910.249	1 782.936	2 742.322
	均值	15 721.4	4 807.902	10 506.127	132 191.72	304 898.91	25 263.798	44 637.04
	标准差	63 147.6	184 367.6	335 444.3	299 582.91	550 649.62	386 586.15	120 087.7
F_6	最值	3 770.079	6 119.393	20 830.748	19 888.732	41 146.717	33 215.100	12 668.32
	均值	4 385.997	6 830.043	25 109.302	25 960.733	46 688.736	40 988.945	15 436.56
	标准差	283.778 1	495.528 65	5 039.633	4 268.600	1 947.267	4 555.865	1 484.988
F_7	最值	54 601.3	144 325.15	206 628.41	192 645.39	469 380.29	583 159.04	782 107.3
	均值	14 245.0	25 504.16	33 411.718	40 247.893	47 007.934	84 204.198	142 352.8
	标准差	43 243.2	96 142.875	83 784.212	17 480.77	586 237.23	14 141 551	4 586 895
F_8	最值	15 713.23	19 186.220	31 516.957	31 589.848	37 352.985	36 694.262	32 989.46
	均值	21 014.04	25 657.004	33 549.035	32 922.861	37 361.356	38 213.174	34 579.26
	标准差	3 786.519	4 731.682	6 231.863	7 026.807	5 623.056	7 575.350	6 658.448
F_9	最值	3 382.896	4 089.011	7 958.084	9 625.405	1 353.901	10 910.131	6 418.969
	均值	3 423.654	4 254.403	9 835.102	11 574.951	13 419.510	13 443.419	6 865.220
	标准差	22.393 02	114.107 29	1 617.806	1 166.977	254.260 02	1 589.567	212.575 1
F_{10}	最值	3 390.009	3 725.611	21 898.128	25 910.188	35 026.325	30 898.447	15 251.35
	均值	3 539.668	3 814.755	26 384.311	29 091.069	35 026.852	32 999.067	17 488.29
	标准差	36.321 33	81.687 642	2 479.008	1 985.004	52.425 367	890.988 64	1 429.347

3.3 10 个基准函数的统计结果

笔者基于 10 个基准函数, 从最优值 (Best)、均值 (Mean) 和标准差 (Std) 3 个维度, 将 ICFOA 与 5 种基础算法进行了对比, 实验结果如表 2 所示。通过表 2 可发现, ICFOA 在大多数函数中表现优异, 几乎都能获得最小的最优值、均值和标准差。对 $F_1 \sim F_6$ 函数, ICFOA 始终能获得理论最优解, 而 CFOA 仅能接近最优解, 表明 ICFOA 比 CFOA 具有更强的开发能力。在 F_2 函数中, ICFOA 表现出

优于其他知名算法的全局能力。对 F_3 函数,虽然 ICFOA 仅找到近似最优解,但其收敛精度仍优于其他算法。针对 F_4 、 F_5 和 F_6 函数,ICFOA 均获得理论最优解。但在 F_4 和 F_9 函数中,ICFOA 的性能逊于 SFO 算法。

考虑到优化算法属于随机算法,笔者采用 Wilcoxon 秩和检验作为补充统计分析方法,以评估结果的统计显著性。值得注意的是,当 $p<0.05$ 时,表明两组数据存在显著差异;当 $p\geq0.05$ 时,表明两组数据差异不显著。同时,笔者采用“NaN”表示两组数据无显著差异。Wilcoxon 秩和检验的结果如表 3 所示。从表 3 可见, F_4 、 F_6 和 F_{10} 函数中均出现“NaN”。这是因为 ICFOA、CFOA 和 SHO 算法的优化结果均达到理论最优解,故 3 组数据差异极小。在其他函数中,ICFOA 算法的结果也与其他算法存在显著差异。Wilcoxon 秩和检验仅能检验算法间的差异,无法评估算法性能。因此,结合表 3 可知,笔者对 CFOA 算法的改进具有显著成效。

表 3 使用 Wilcoxon 秩和检验的算法在 10 个基准函数上的统计结果

Tab. 3 Statistical results of the algorithm using the Wilcoxon rank-sum test on 10 benchmark functions

函数	ICFOA vs.					
	CFOA	ROA	AOA	SCA	SFO	SHO
F_1	3.34×10^{-1}	1.21×10^{-12}	1.21×10^{-12}	1.21×10^{-12}	1.21×10^{-12}	1.21×10^{-12}
F_2	1.21×10^{-12}	1.21×10^{-12}	1.21×10^{-12}	1.21×10^{-12}	1.21×10^{-12}	1.21×10^{-12}
F_3	4.57×10^{-12}	1.21×10^{-12}	1.21×10^{-12}	1.21×10^{-12}	1.21×10^{-12}	1.21×10^{-12}
F_4	NaN	1.21×10^{-12}	1.21×10^{-12}	NaN	1.21×10^{-12}	1.21×10^{-12}
F_5	4.83×10^{-1}	1.07×10^{-7}	3.02×10^{-11}	3.02×10^{-11}	1.21×10^{-12}	1.21×10^{-12}
F_6	4.62×10^{-10}	3.02×10^{-11}	3.02×10^{-11}	3.02×10^{-11}	1.21×10^{-12}	NaN
F_7	6.10×10^{-3}	3.11×10^{-1}	7.98×10^{-2}	3.02×10^{-11}	1.21×10^{-12}	1.21×10^{-12}
F_8	1.07×10^{-7}	3.02×10^{-11}	1.22×10^{-2}	3.02×10^{-11}	1.21×10^{-12}	1.21×10^{-12}
F_9	1.21×10^{-12}	4.57×10^{-12}	1.22×10^{-12}	1.21×10^{-12}	1.21×10^{-12}	1.21×10^{-12}
F_{10}	NaN	1.21×10^{-12}	NaN	1.21×10^{-12}	1.21×10^{-12}	1.21×10^{-12}

3.4 ICFOA 与其他算法的收敛性分析

收敛曲线是评估算法性能的重要指标之一。图 2 给出了这些算法在若干基准函数上的收敛曲线。

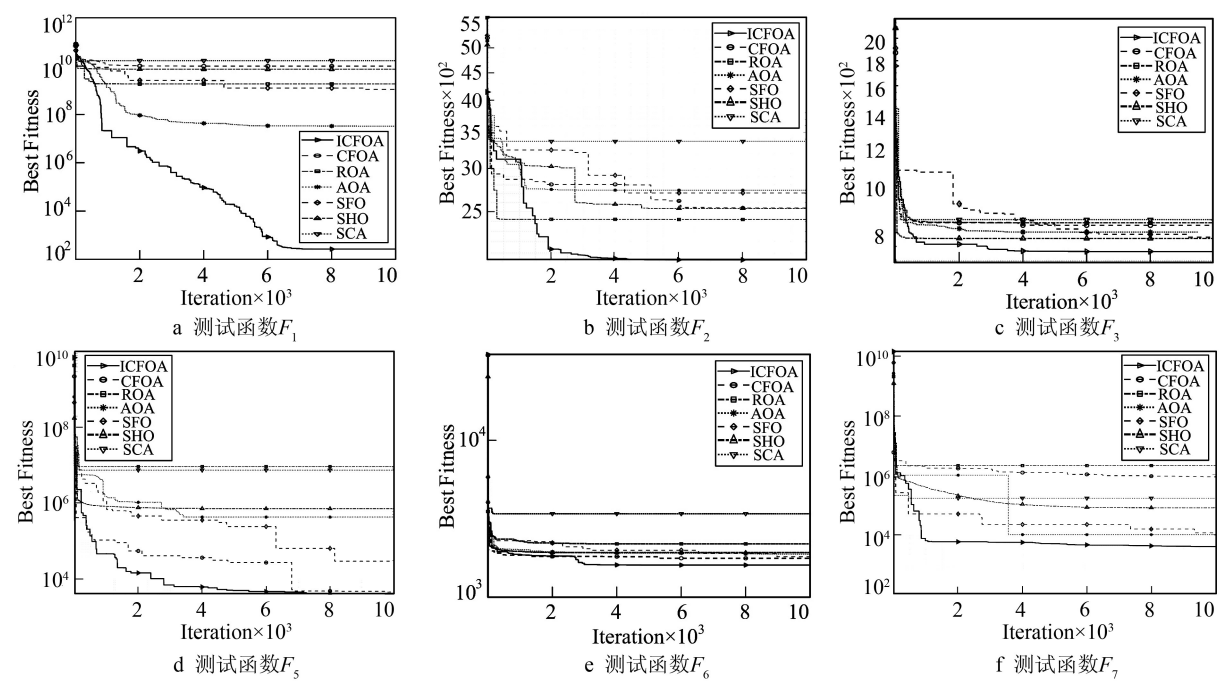


图 2 ICFOA 收敛性及部分函数的比较算法

Fig. 2 Comparison of convergence and partial function algorithms for ICFOA

从图2可看出, ICFOA 算法在单峰函数 F_1 和 F_2 上表现出显著提升的收敛速度。对 F_5 函数, 虽然 ICFOA 的效果并非最佳, 但其与最优解仍存在微小差异。由此可知 ICFOA 具备强大的探索能力。在 F_6 和 F_7 函数中, ICFOA 与 ROA 算法表现相近, 但 ICFOA 收敛速度更快。 F_9 函数中, ICFOA、CFOA 和 SCA 算法均能快速收敛。 综上, ICFOA 在各类函数中均表现出卓越的收敛能力。

4 光伏电站 MPPT 控制优化

算法经过优化后可应用于光伏电站发电测试, 模型如图3所示, 其主要用于不同算法在光照变化下寻找最大功率点的差值。在电路中, 光伏阵列的输出电压、电流先通过转换器转换为数字信号, 传输给 MPPT(Maximum Power Point Tracking) 模块; 并且其依据采集的电参数, 通过 ICFOA 算法计算出最大功率点对应的控制指令, 再传递给 PWM (Pulse Width Modulation) 发生器, 生成对应占空比的 PWM 信号, 驱动 Boost 变换器, 最终实现光伏阵列输出功率的最大化跟踪控制。表4给出了 ICFOA、CFOA、ROA、SCA、AOA、WOA 及 SFO 在光伏电站发电测试模型中的对比结果。

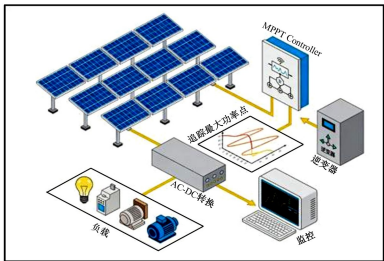


图3 光伏电站发电测试模型
Fig.3 Photovoltaic power plant generation testing model

表4 不同优化算法的比较结果

Tab.4 Comparison results of different optimization algorithms

算法	ICFOA	CFOA	ROA	SCA	AOA	WOA	SFO
RE/%	0.014 3	0.162 2	0.026 1	0.018 4	0.055 9	0.031 7	0.028 8
	0.036 2	2.690 8	0.038 9	0.041 7	0.041 1	0.048 2	0.047 3
	0.019 6	1.136 2	2.743 4	0.039 6	0.154 3	0.246 4	0.997 5
	0.005 7	1.646 7	5.023 7	0.076 6	8.386 7	0.114 5	0.279 4
	0.024 5	1.587 4	6.539 6	0.056 4	3.981 8	0.467 1	1.863 5
RMSE	0.067 4	35.483 2	27.867 2	24.366 1	21.865 2	0.087 3	15.062 1
	0.354 3	4.140 1	0.531 8	0.418 4	0.369 3	0.498 4	0.531 8
	0.160 3	21.854 4	43.143 7	22.476 8	0.395 5	1.654 4	3.343 8
	0.348 8	9.136 4	12.312 5	10.854 3	0.919 1	34.470 9	20.504
	0.228 9	29.566 7	23.009 7	3.910 1	6.014 4	3.186 5	15.875

5 结 语

笔者在 CFOA 算法的基础上, 引入了个性化捕鱼策略(PFS), 并提出改进型 CFOA(ICFOA)。虽然 CFOA 已被广泛应用于解决各类设计问题, 但 ICFOA 仍存在易陷入算法停滞和局部最优等缺陷。为此, 采用 PFS 策略提升其优化能力。即其基于 PFS 增强了全局搜索能力, 对基础 CFOA 摆脱局部最优值具有显著影响。为验证 ICFOA 的有效性, 通过 10 个基准函数将其与其他 5 种知名算法进行对比。结果表明, ICFOA 在大多数基准函数及实际问题中均表现优异。

参考文献:

[1] 黄东海. 基于多策略改进的黏菌优化算法及其应用 [D]. 南宁: 广西大学, 2025.
HUANG D H. Multi-Strategy Improvement Based Optimisation Algorithm for Slime Moulds and Its Application [D]. Nanning: Guangxi University, 2025.

[2] 吴莉娟, 吕莉, 肖人彬, 等. 面向约束优化问题的聚类多目标狼群算法 [J/OL]. 信息与控制, 2025: 1-17 [2025-12-29]. <https://doi.org/10.13976/j.cnki.xk.2024.4511>.

WU L J, LÜ L, XIAO R B, et al. Clustering Multi-Objective Wolf Pack Algorithm for Constrained Optimization Problems [J/OL]. Information and Control, 2025: 1-17 [2025-12-29]. <https://doi.org/10.13976/j.cnki.xk.2024.4511>.

- [3] 易进伟. 基于蛱螂优化算法的多策略改进及应用 [D]. 赣州: 江西理工大学, 2025.
YI J W. Multi-Strategy Enhancements and Applications of the Dung Beetle Optimization Algorithm [D]. Ganzhou: Jiangxi University of Science and Technology, 2025.
- [4] 杨程峰. 基于改进白鲸优化算法的微电网优化调度研究 [D]. 吉林: 东北电力大学, 2025.
YANG C F. Optimal Scheduling Study of Microgrid Based on Improved Beluga Optimisation Algorithm [D]. Jilin: Northeast Electric Power University, 2025.
- [5] 徐勇, 赵文贺. 基于多种策略改进的小龙虾优化算法 [J]. 信息技术与信息化, 2025(10): 36-40.
XU Y, ZHAO W H. An Improved Crayfish Optimization Algorithm Based on Multiple Strategies [J]. Information Technology and Informatization, 2025(10): 36-40.
- [6] 王家铭, 朱金荣. 基于小龙虾优化算法的模糊双闭环控制的直流微电网母线稳压策略 [J]. 电工技术, 2025(10): 173-176.
WANG J M, ZHU J R. Fuzzy Dual Closed-Loop Control Based on Crayfish Optimization Algorithm for DC Microgrid Bus Voltage Stabilization Strategy [J]. Electrical Engineering, 2025(10): 173-176.
- [7] 谢军飞, 张海清, 李代伟, 等. 基于惯性权重的改进鲸鱼优化算法 [J]. 成都信息工程大学学报, 2025, 40(5): 600-604.
XIE J F, ZHANG H Q, LI D W, et al. An Improved Whale Optimization Algorithm Based on Inertia Weight [J]. Journal of Chengdu University of Information Technology, 2025, 40(5): 600-604.
- [8] 谢良波, 韩伸, 张钰坤. 融合邻域搜索的自适应鲸鱼优化算法 [J]. 电子测量与仪器学报, 2024, 38(12): 124-134.
XIE L B, HAN S, ZHANG Y K. Adaptive Whale Optimization Algorithm Combining Neighborhood Search [J]. Journal of Electronic Measurement and Instrumentation, 2024, 38(12): 124-134.
- [9] 王钰霖, 孙丽颖. 局部阴影下基于改进灰狼算法的光伏 MPPT [J]. 辽宁工业大学学报(自然科学版), 2025, 45(2): 116-120.
WANG Y L, SUN L Y. PV MPPT Based on Improved Grey Wolf Optimization Algorithm under Partial Shading Conditions [J]. Journal of Liaoning University of Technology (Natural Science Edition), 2025, 45(2): 116-120.
- [10] 蔡宇航. 灰狼优化算法的改进及其应用研究 [D]. 无锡: 江南大学, 2025.
CAI Y H. Research on the Improvement and Application of Grey Wolf Optimization Algorithm [D]. Wuxi: Jiangnan University, 2025.
- [11] JIA H, WEN Q, WANG Y, et al. Catch Fish Optimization Algorithm: A New Human Behavior Algorithm for Solving Clustering Problems [J]. Cluster Computing, 2024, 27: 13295-13332.
- [12] 李雨恒, 高尚, 孟祥宇. 基于精英引导的改进哈里斯鹰优化算法 [J]. 计算机工程与科学, 2024, 46(2): 363-373.
LI Y H, GAO S, MENG X Y. An Improved Harris Hawks Optimization Algorithm Based on Elite Guidance [J]. Computer Engineering & Science, 2024, 46(2): 363-373.
- [13] HSIEH C H, ZHANG Q, XU Y, et al. CMAIS-WOA: An Improved WOA with Chaotic Mapping and Adaptive Iterative Strategy [J/OL]. Discrete Dynamics in Nature and Society, 2023(1): 8160121 [2026-04-24]. <https://doi.org/10.1155/2023/8160121>.
- [14] KUMAR A, WU G, ALI M Z, et al. A Test-Suite of Non-Convex Constrained Optimization Problems from the Real-World and Some Baseline Results [J/OL]. Swarm and Evolutionary Computation, 2020, 56: 100693 [2026-04-24]. <https://doi.org/10.1016/j.swevo.2020.100693>.
- [15] SI Q, LI C. Indoor Robot Path Planning Using an Improved Whale Optimization Algorithm [J/OL]. Sensors, 2023, 23(8): 3988 [2026-04-24]. <https://doi.org/10.3390/s23083988>.
- [16] CAI Y, GUO C, CHEN X. An Improved Sand Cat Swarm Optimization with Lens Opposition-Based Learning and Sparrow Search Algorithm [J/OL]. Scientific Reports, 2024, 14(1): 20690 [2026-04-24]. <https://doi.org/10.1038/s41598-024-71581-2>.

(责任编辑: 刘俏亮)