

DOI: 10.11992/tis.201706091

网络出版地址: <http://kns.cnki.net/kcms/detail/23.1538.tp.20171109.1255.028.html>

# 具有 Levy 变异和精英自适应竞争机制的蚁狮优化算法

景坤雷<sup>1,2</sup>, 赵小国<sup>1,2,3</sup>, 张新雨<sup>1,2</sup>, 刘丁<sup>1,2</sup>

(1. 西安理工大学 晶体生长设备及系统集成国家地方联合工程研究中心, 陕西 西安 710048; 2. 陕西省复杂系统控制与智能信息处理重点实验室, 陕西 西安 710048; 3. 西安建筑科技大学 机电工程学院, 陕西 西安 710055)

**摘 要:** 针对蚁狮优化算法易陷入局部最优、收敛速度慢的缺点, 本文提出一种具有 Levy 变异和精英自适应竞争机制的蚁狮优化算法。利用服从 Levy 分布的随机数对种群较差个体进行变异, 可改善种群多样性提高算法的全局搜索能力; 精英自适应竞争机制使得多个精英并行带领种群寻优, 提高了算法的收敛速度, 为避免较大计算量, 并行竞争的精英个数会随着寻优代数增加而减少。同多个改进算法进行比较, 结果表明本文所提算法具有更好的寻优精度和收敛速度。最后将本文改进算法应用于硅单晶热场温度模型的参数辨识, 仿真结果说明该算法具有较好的参数辨识能力。

**关键词:** 蚁狮优化算法; Levy 变异; 精英自适应竞争; 收敛速度; 硅单晶; 参数辨识

**中图分类号:** TP18 **文献标志码:** A **文章编号:** 1673-4785(2018)02-0236-07

中文引用格式: 景坤雷, 赵小国, 张新雨, 等. 具有 Levy 变异和精英自适应竞争机制的蚁狮优化算法[J]. 智能系统学报, 2018, 13(2): 236-242.

英文引用格式: JING Kunlei, ZHAO Xiaoguo, ZHANG Xinyu, et al. Ant lion optimizer with levy variation and adaptive elite competition mechanism[J]. CAAI transactions on intelligent systems, 2018, 13(2): 236-242.

## Ant lion optimizer with levy variation and adaptive elite competition mechanism

JING Kunlei<sup>1,2</sup>, ZHAO Xiaoguo<sup>1,2,3</sup>, ZHANG Xinyu<sup>1,2</sup>, LIU Ding<sup>1,2</sup>

(1. National & Local Joint Engineering Research Center of Crystal Growth Equipment and System Integration, Xi'an University of Technology, Xi'an 710048, China; 2. Shaanxi Key Laboratory of Complex System Control and Intelligent Information Processing, Xi'an 710048, China; 3. School of Mechanical and Electrical Engineering, Xi'an University of Architecture and Technology, Xi'an 710055, China)

**Abstract:** The ant lion optimizer (ALO) reveals such deficiencies as easily relapsing into local optimum and low convergence speed. This paper proposed an improved ALO algorithm with Levy variation and adaptive elite competition mechanism. By carrying out Levy variation to poor individuals, the diversity of population and the global search ability of the algorithm can be increased. Moreover, the adaptive elite competition mechanism that many elites lead the population to search at the same time can improve the convergence speed of the algorithm. To reduce the amount of calculation, the number of the elites competing in parallel will decrease with the increase of iterations. By contrast with other improved optimization algorithms, the test results show that the improved algorithm proposed in this paper has better search precision and convergence speed. Finally, this improved algorithm is applied to identify parameters of silicon single crystal thermal field temperature model and the simulation results prove its excellent ability of parameters identification.

**Keywords:** ant lion optimizer; Levy variation; adaptive elite competition; convergence speed; parameters identification;

收稿日期: 2017-06-30. 网络出版日期: 2017-11-09.

基金项目: 国家自然科学基金重点项目 (61533014); 陕西省教育厅专项科研计划项目 (17JK0456).

通信作者: 刘丁. E-mail: [liud@xaut.edu.cn](mailto:liud@xaut.edu.cn).

近几十年来, 越来越多的群体智能算法被提出, 并以其操作简单、不受求解对象约束等特点在工程领域得到广泛应用。蚁狮优化算法 (ant lion

optimizer, ALO) 是澳大利亚学者 Seyedali Mirjalili<sup>[1]</sup>通过研究蚁狮捕食蚂蚁的仿生学机制,提出的一种智能优化算法。ALO 算法以其调节参数少、求解精度高的优点,备受科研工作者的青睐,目前已被成功应用于天线布局优化、分布式系统的选址和控制器增益值优化等工程领域<sup>[2-4]</sup>。

文献[5]为改善种群多样性,通过对基本算法的精英化过程引入了权值操作,提出了 MALO 算法,但是随着种群不断向精英靠拢,多样性仍会不可避免地降低,因而该方法没有从根本上提高算法的全局搜索能力;文献[6]为减小适应值较差个体对种群的误导,提出一种具有混沌侦查机制的 CIALO 算法,提高了种群对求解域的映射能力,但并未改善算法的收敛速度。

针对以上不足,本文提出一种具有 Levy 变异和精英自适应竞争机制的蚁狮优化算法 (ALO with Levy variation and adaptive Elite competition, LEALO)。服从 Levy 分布的随机数具有短距离游走结合偶尔长距离跳跃的特征,利用其对种群较差的个体进行变异,可以改善种群多样性,实现对求解域的充分探索,从而提高算法的全局搜索能力;多个精英之间的自适应并行竞争,有助于种群更快地锁定更优的区域,保证算法收敛速度的同时避免了较大的计算量。本文选择标准函数对 LEALO 算法进行测试,并与基本 ALO 算法<sup>[1]</sup>、MALO 算法<sup>[5]</sup>和 CIALO<sup>[6]</sup>算法进行比较,结果表明 LEALO 算法具有更高的寻优精度和收敛速度。最后将其用于硅单晶热场温度模型的参数辨识中,仿真结果说明了该算法良好的优化能力。

## 1 蚁狮优化算法及其缺点

### 1.1 ALO 算法原理

蚁狮是一种靠捕食蚂蚁生存的蚁蛉科昆虫,以其独特的狩猎方式得名。蚁狮“狩猎”时先在沙地上挖出“陷阱”,然后躲入穴底等待“猎物”,一旦蚂蚁进入“陷阱”,为防止其逃走蚁狮会立刻向外刨出沙土使其滑入穴底进而捕食。Mirjalili 将二者间的仿生学机制公式化再现,提出了蚁狮优化算法。该算法的主要步骤如下。

#### 1.1.1 蚁狮修筑陷阱

根据适应值,通过轮盘赌操作从上一代的蚂蚁种群中选择个体。被选中的个体将和精英一起作为蚁狮修筑“陷阱”。

#### 1.1.2 蚂蚁随机游走

按照式 (1) 产生随机游走的蚂蚁种群:

$$X(t) = [0, \text{cumsum}(2r(t_1) - 1), \text{cumsum}(2r(t_2) - 1), \dots, \text{cumsum}(2r(t_n) - 1)] \quad (1)$$

式中: cumsum 为蚂蚁游走位置的累积,  $n$  是设置的最大迭代次数,  $t$  为游走的步数,  $r(t)$  为随机数 0 或 1。为了保证蚂蚁在求解空间内游走,需对其爬行位置进行标准化:

$$X_i^t = \frac{(X_i^t - a_i) \times (d_i^t - c_i^t)}{(b_i - a_i)} + c_i^t \quad (2)$$

式中:  $a_i$  和  $b_i$  为游走过程中第  $i$  个变量的最小值和最大值,  $c_i^t$  和  $d_i^t$  为第  $t$  代第  $i$  个变量的最小值和最大值。

#### 1.1.3 蚂蚁进入陷阱

蚂蚁爬入陷阱的过程,可以看作蚂蚁围绕修筑“陷阱”的蚁狮游走,即蚂蚁游走的区域边界受蚁狮位置的影响:

$$\begin{cases} c_i^t = \text{Antlion}_j^t + c^t \\ d_i^t = \text{Antlion}_j^t + d^t \end{cases} \quad (3)$$

式中:  $c^t$  和  $d^t$  分别为第  $t$  代所有变量中的最小值和最大值,  $c_i^t$  和  $d_i^t$  分别是第  $i$  只蚂蚁的所有位置里的最小值和最大值,  $\text{Antlion}_j^t$  是第  $t$  代选中的第  $j$  个蚁狮。

#### 1.1.4 蚂蚁滑落穴底

一旦蚂蚁进入陷阱,为阻止其逃走,蚁狮会立即向穴外刨出沙土使其滑入穴底。该过程可以看作蚂蚁绕蚁狮游走的半径在不断缩小:

$$\begin{cases} c^t = \frac{c^t}{I} \\ d^t = \frac{d^t}{I} \end{cases} \quad (4)$$

式中:  $I = +10^w \cdot t/T$ ,  $T$  为最大迭代次数 ( $w=2, t>0.1T$ ;  $w=3, t>0.5T$ ;  $w=4, t>0.75T$ ;  $w=5, t>0.9T$ ;  $w=6, t>0.95T$ )。

#### 1.1.5 蚁狮重筑陷阱

若游走的蚂蚁种群中出现了适应值高于蚁狮的个体,则该个体为新的精英。即该个体将作为蚁狮在下一代修筑“陷阱”:

$$\text{Antlion}_j^t = \text{Ant}_i^t, f(\text{Ant}_i^t) < f(\text{Antlion}_j^t) \quad (5)$$

式中:  $t$  为当前代数,  $\text{Ant}_i^t$  是第  $t$  代适应值最佳的第  $i$  个蚂蚁,  $f$  为适应值函数。

#### 1.1.6 蚂蚁种群精英化

绕精英游走的蚂蚁种群,影响着绕轮盘赌选择的个体游走的蚂蚁种群。

$$\text{Ant}_i^t = \frac{R_A^t + R_E^t}{2} \quad (6)$$

式中:  $R_A^t$  表示绕轮盘赌选中的蚁狮游走的蚂蚁种群,  $R_E^t$  表示绕精英蚁狮游走的蚂蚁种群。

### 1.2 ALO 算法缺点

在 ALO 算法中,蚂蚁种群绕精英蚁狮的随机游走保证了寻优过程的收敛性,轮盘赌操作在一定程度上有助于提高蚂蚁种群的全局搜索能力。但

是,算法仍存在以下问题:蚁狮的捕食半径的随着迭代次数的增加阶段性收缩,会导致种群多样性的逐渐降低,算法一旦陷入局部最优就难以跳出;若当代精英和轮盘赌选择的个体并不处于全局最优区域时,整个种群在单个精英带领下会降低算法的收敛速度。

## 2 改进算法 (LEALO)

针对 ALO 算法存在的缺点,本文引入 Levy 变异和并行的自适应精英竞争机制。将服从 Levy 分布的随机数用于种群较差个体的变异,可以有效改善种群的多样性,提高算法的全局搜索能力;多个精英同时带领种群探索加快了算法的收敛速度。另外,引入精英的个数随迭代次数的增加而减少的自适应机制可以避免较大的计算量。

### 2.1 Levy 变异机制

Levy 游走一词是由法国数学家保罗·列维提出的,而后有学者发现很多生物群体的活动方式均可以用 Levy 游走模式进行描述<sup>[7]</sup>。研究人员通过对生物群体基于 Levy 游走模式的活动方式进行研究,形成了一种 Levy 飞行觅食假说,即 Levy 游走可以提高觅食效率,基于 Levy 游走的觅食方式自然适应性更强。Levy 飞行表现为长期短距离游走和偶尔长距离跳跃的结合,这种长距离跳跃具有方向多变性的特点<sup>[8]</sup>。

利用 Levy 飞行特点形成 Levy 变异机制来提高种群的多样性,保证了种群对附近区域详细搜索的同时又具有一定的突变性。两种方式交替从而实现求解域的充分遍历,有助于提高算法的全局搜索能力<sup>[9]</sup>。Levy 飞行服从 Levy 分布,其概率密度函数如下:

$$P_{\alpha,\gamma}(z) = \frac{1}{\pi} \int_0^{\infty} \exp(-\gamma q^{\alpha}) \cos(qz) dq \quad (7)$$

式中:  $0 < \alpha \leq 2$  控制 Levy 分布图形的锐度,  $\gamma > 0$  控制分布的尺度。当  $\alpha = 2$  时, Levy 分布等同于高斯分布,  $\alpha = 1$  时, Levy 分布等同于柯西分布。对于一般的  $\alpha$  取值,通过 Levy 分布的概率密度函数分析较为困难,所以通常采用模拟方法产生 Levy 分布的随机数<sup>[10-12]</sup>。为了使 Levy 飞行轨迹充分遍历设定的求解域,对文献<sup>[13]</sup>的模拟公式做修改如式(8):

$$\begin{cases} \delta_{\mu} = \left\{ \frac{\Gamma(1+\beta) \sin(\pi\beta/2)}{\Gamma[(1+\beta)/2] 2^{(\beta-1)/2} \beta} \right\}^{1/\beta} \\ \delta_v = 1 \end{cases} \quad (8)$$

$$\begin{cases} \mu \sim N(0, \delta_{\mu}^2) \\ v \sim N(0, \delta_v^2) \end{cases} \quad (9)$$

$$\begin{cases} S = \frac{\mu}{|v|^{1/\beta}} \\ \lambda_i = \text{cumsum}(S_i) \end{cases} \quad (10)$$

式中:  $i = 1, 2, \dots, n$  为寻优的维度,  $\text{cumsum}(S_i)$  为前  $i$  次 Levy 飞行位置的累积。

$$L(\lambda_i) = \begin{cases} l_b, & \lambda_i < l_b \\ a \cdot \lambda_i, & l_b < \lambda_i < u_b \\ u_b, & \lambda_i > u_b \end{cases} \quad (11)$$

根据式(8)~(10)计算得到 Levy 飞行轨迹,式(11)通过尺度因子  $a$  和限幅操作将 Levy 飞行位置映射在求解域  $[l_b, u_b]$  内。取  $a = 0.5, l_b = -5, u_b = 5$  模拟出二维的 Levy 飞行轨迹如图 1 所示,充分验证了 Levy 飞行短距离结合偶尔长距离跳跃的特征,对求解域实现了充分的探索。因此选择适当数量的蚁狮种群进行 Levy 变异可以显著改善整个种群的多样性,从而提高算法的全局搜索能力。

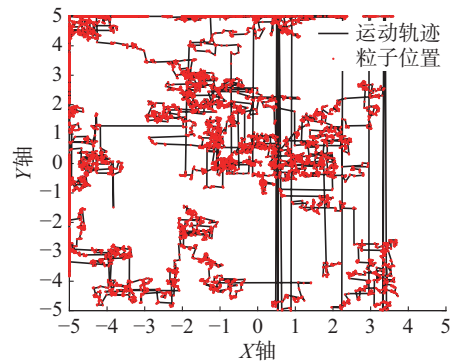


图1 设定搜索范围内的 Levy 飞行轨迹图

Fig. 1 Simulation tracks of Levy flights in the set region

计算出 Levy 飞行路径后  $L(\lambda)$ , 选择适应值较差的若干个体  $x_i$  按照式(12)进行变异操作:

$$x'_i = L(\lambda_i) \quad (12)$$

### 2.2 精英自适应竞争机制

单个精英所拥有的极值信息极其有限,因此有必要建立精英库存储历代较佳的个体(变异后适应值较佳的个体也会被存入精英库)。对 ALO 算法引入精英竞争机制,在每一代的寻优中,多个精英之间并行竞争,而不是通过轮盘赌的方式选择。在多个精英的同时带领下,种群能够快速锁定相对较优解的所在区域,有助于加快算法收敛速度<sup>[14-15]</sup>。

为保证寻优前期的收敛速度,应选取较多个精英参与竞争;而后期,应减少精英个数避免较大的计算量。因此并行竞争的精英个数应随着迭代次数的增加而衰减。这种自适应选取方式在保证算法寻优速度的同时避免了不必要的计算量。设置精英个数范围为  $[n_{\min}, n_{\max}]$ , 则对于当代精英个数  $n(t)$  和迭代次数  $t$ , 构造如下关系式:

$$\begin{cases} n(t) = \text{round}\left(\frac{n_{\min}}{1 + \left(\frac{n_{\min}}{n_{\max}} - 1\right)h(t)}\right) \\ h(t) = 1 - \left(\frac{t}{T}\right)^2 \end{cases} \quad (13)$$

式中: round()为取整函数,  $T$ 为设置的最大迭代次数。设置精英个数的范围为[1,5]时,随迭代次数的增加,精英个数变化曲线如图2所示。

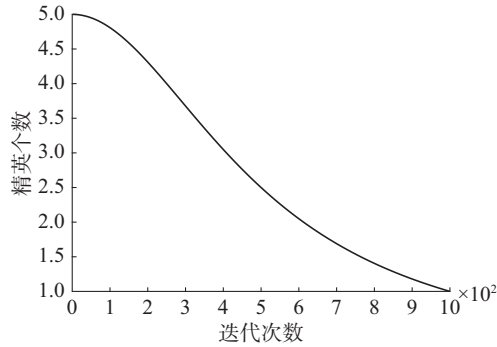


图2 精英个数和迭代次数之间的函数曲线

Fig. 2 Function curve between the number of elites and iterations

建立容量为 $n$ 的精英库,每代寻优完成后对其更新,使得库内始终存放的是截止当前代为止最优的 $n$ 个个体。根据式(13)确定精英个数,蚂蚁种群按照式(1)、(2)绕多个精英游走,式(3)变为

$$\begin{aligned} c'_i &= \text{Elite}'_j + c'_i, d'_i = \text{Elite}'_j + d'_i \\ i &= 1, 2, \dots, n \times N; j = 1, 2, \dots, n \end{aligned} \quad (14)$$

式中:  $\text{Elite}'_j$ 为来自精英库的精英,  $N$ 为设定的种群数目。式(4)不作变化,式(6)中的 $\text{Ant}'_i$ 代表被 $n$ 个精英精英化后的蚂蚁种群,即 $i = 1, 2, \dots, n \times N$ 。式(5)变为

$$\sum_{i=1}^{n \times N} f_{\text{sort}}(\text{Ant}'_i) \Rightarrow \sum_{i=1}^n \text{Antlion}'_i{}^{t+1} \quad (15)$$

式中:  $f_{\text{sort}}$ 为排序函数,  $\Rightarrow$ 表示提取排序后的前 $n$ 个较优个体作为下一代蚁狮种群。

### 2.3 LEALO 算法执行伪代码

Algorithm: LEALO algorithm

$N$ —The number of searching individuals

$T$ —The max iters

$n$ —The number of selected elites in elites library

Initialize Antlions position

Building Elites library

while the max iters  $T$  is not met

    Get  $n$  using Eqs.(13)

    for every elite  $i = 1, 2, \dots, n$

        for every ant  $i = 1, 2, \dots, N$

            Select a antlion from the Antlions;

            Ants walk around  $i$ th elite using Eq.(1)

            Normalize ants using Eq.(2);

            Update ants using Eqs.(14)(4);

            Elitism the ants using Eqs.(6);

            Calculate the objective values of ants;

            Update the Elites library using Eq.(15);

        end for

    end for

    Make Levy-mutation using Eq.(8)(9)(10)

(11)(12)

end while

## 3 测试对比分析与应用

### 3.1 仿真实验

下面通过6个标准函数测试 LEALO 算法的寻优精度和收敛速度,测试函数设置见表1。其中 $f_1$ 、 $f_2$ 为单峰函数, $f_3 \sim f_6$ 为多峰函数。

表1 标准测试函数

Table 1 Standard test functions

测试函数	维度/种群/迭代次数	函数表达式	搜索范围	理论最优解	精度要求
$f_1$ (Sphere)	30, 30, 1 000	$\sum_{i=1}^n x_i^2$	$[-100, 100]$	$f_{\min}(0, \dots, 0) = 0$	$10^{-10}$
$f_2$ (Schwefel)	10, 30, 1 000	$\sum_{i=1}^n  x_i  + \prod_{i=1}^n  x_i $	$[-10, 10]$	$f_{\min}(0, \dots, 0) = 0$	$10^{-6}$
$f_3$ (Eggcrate)	2, 30, 1 000	$x^2 + y^2 + 25(\sin^2 x + \sin^2 y)$	$[-10, 10]$	$f_{\min}(0, 0) = 0$	$10^{-15}$
$f_4$ (Salomon)	10, 30, 1 000	$-\cos\left(2\pi\sqrt{\sum_{i=1}^n x_i^2}\right) + 0.1\sqrt{\sum_{i=1}^n x_i^2} + 1$	$[-5, 5]$	$f_{\min}(0, \dots, 0) = 0$	$10^{-8}$
$f_5$ (Griewank)	10, 30, 3 000	$\frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	$[-600, 600]$	$f_{\min}(0, \dots, 0) = 0$	$10^{-9}$
$f_6$ (Ackley)	10, 30, 3 000	$\exp\left(4\sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2} - \frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)\right) + 20 + e$	$[-32, 32]$	$f_{\min}(0, \dots, 0) = 0$	$10^{-10}$



仿真平台, 操作系统: win7 旗舰版 (64 b); CPU: Intel(R)Core(TM)i5-4590; 主频: 3.30 GHz; RAM: 4.00 GB; 编程工具: MATLAB2016b。选择 3 种算法 (ALO、MALO、CIALO) 和 LEALO 算法进行对比, 测试 100 次, 分别统计历代最优解、平均解和寻

优成功率 (当寻优值达到设置精度时, 视作寻优成功)。因 3 种改进算法效果均优于基本 ALO 算法 (结果见图 3), 考虑到表格篇幅, 只给出 3 种改进算法的测试结果, 如表 2 所示。

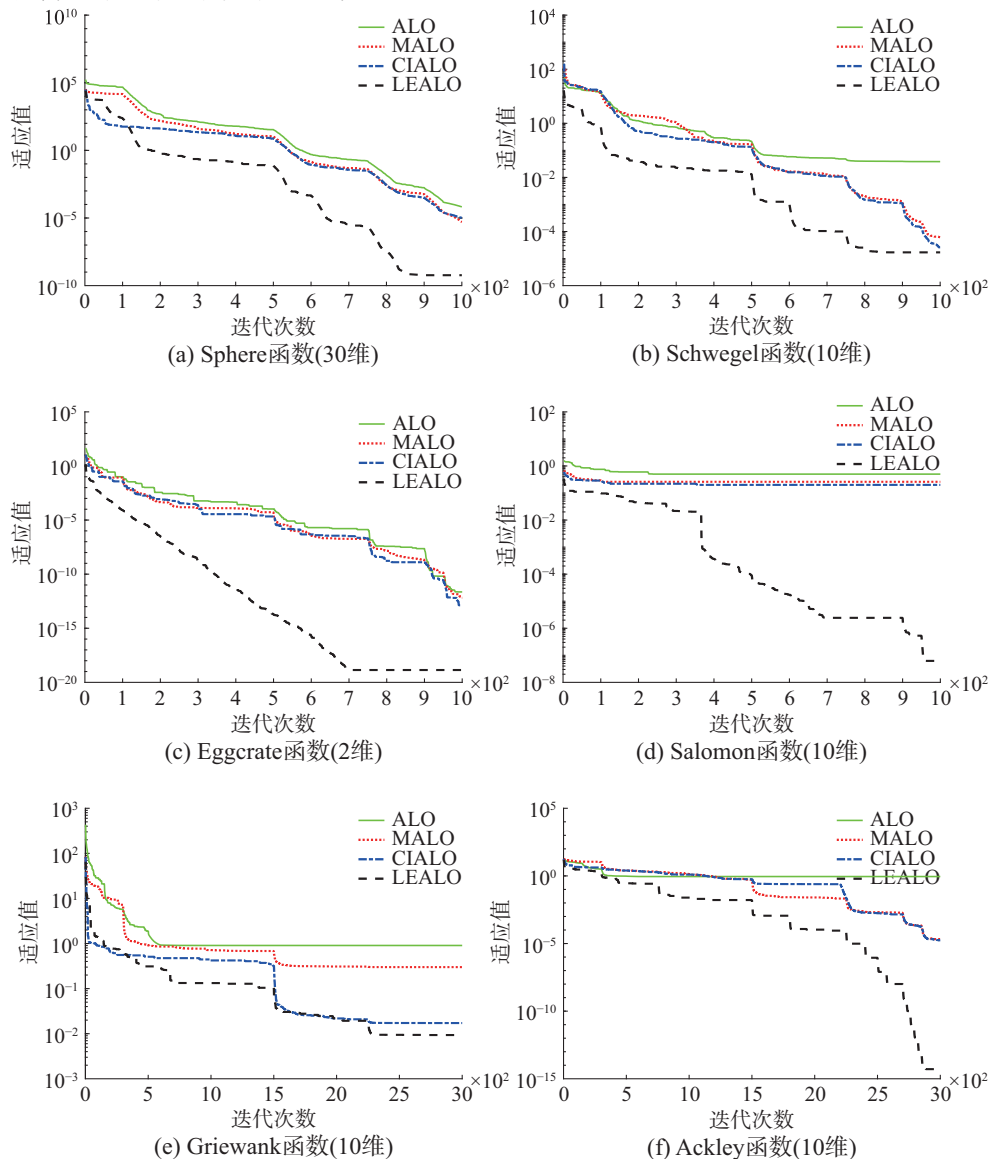


图 3 对数坐标下 3 种算法的寻优收敛曲线

Fig. 3 Convergence curves of the three algorithms under logarithmic coordinates

表 2 LEALO、CIALO、MALO 3 种改进优化算法的测试结果对比

Table 2 Test results comparison of LEALO, CIALO and MALO

算法函数	历代最优解			平均最优解			寻优成功率/%		
	LEALO	CIALO	MALO	LEALO	CIALO	MALO	LEALO	CIALO	MALO
$f_1$	$1.866\ 3 \times 10^{-11}$	$6.625\ 2 \times 10^{-6}$	$2.081\ 3 \times 10^{-6}$	$5.885\ 2 \times 10^{-10}$	$1.013\ 8 \times 10^{-5}$	$5.121\ 3 \times 10^{-6}$	80	0	0
$f_2$	$5.453\ 9 \times 10^{-8}$	$1.155\ 3 \times 10^{-5}$	$1.140\ 8 \times 10^{-5}$	$1.697\ 0 \times 10^{-5}$	$2.601\ 5 \times 10^{-5}$	$6.170\ 5 \times 10^{-5}$	60	0	0
$f_3$	$3.381\ 2 \times 10^{-20}$	$4.457\ 8 \times 10^{-14}$	$6.152\ 5 \times 10^{-14}$	$1.379\ 0 \times 10^{-19}$	$1.328\ 2 \times 10^{-13}$	$6.494\ 6 \times 10^{-13}$	100	20	0
$f_4$	$3.344\ 8 \times 10^{-8}$	0.199 9	0.199 9	$6.215\ 1 \times 10^{-8}$	0.199 9	0.259 9	100	0	0
$f_5$	$4.774\ 3 \times 10^{-9}$	$6.002\ 2 \times 10^{-9}$	0.076 3	0.009 3	0.017 2	0.071 0	80	60	0
$f_6$	$4.440\ 9 \times 10^{-15}$	$1.136\ 3 \times 10^{-5}$	$1.428\ 3 \times 10^{-5}$	$5.151\ 4 \times 10^{-15}$	$1.724\ 0 \times 10^{-5}$	$2.011\ 5 \times 10^{-5}$	100	0	0

精度方面:对于函数 $f_1$ ,MALO 算法高于 CIALO 算法;对于函数 $f_2$ 、 $f_3$ 、 $f_4$ 、 $f_5$ 、 $f_6$ ,MALO 算法低于 CIALO 算法。寻优成功率方面:对于函数 $f_3$ 、 $f_5$ ,CIALO 算法高于 MALO 算法。整体而言,这两种算法的寻优精度和成功率均不高,而 LEALO 算法在寻优精度和寻优成功率上均远优于 MALO 算法和 CIALO 算法。

为了使寻优精度和收敛速度对比更为直观,分别取经过 100 次测试的 4 种算法收敛曲线平均值,绘制在对数坐标系中,如图 3 所示。对于函数 $f_1$ 、 $f_3$ 、 $f_4$ 、 $f_6$ ,LEALO 算法的寻优精度远高于 MALO 和 CIALO 算法;对于函数 $f_2$ 、 $f_5$ ,LEALO 算法寻优精度略高于和 MALO 和 CIALO 算法,但是收敛速度明显快于这两种算法;对于函数 $f_4$ 、 $f_5$ ,MALO 和 CIALO 算法均早早陷入局部最优不再收敛,而 LEALO 算法能够不断跳出局部最优对求解域进行充分探索,在得到更高精度解的同时保证了较快的收敛速度。综合表 2 和图 3 得出,LEALO 算法有效克服了基本 ALO 算法及 MALO 和 CIALO 两种改进算法易陷入局部最优、收敛速度慢的缺点,对于高维度变量的多峰函数可以实现高精度快速寻优。

### 3.2 硅单晶热场温度模型参数的辨识

硅单晶是一种重要的半导体材料,基于直拉法的单晶炉是制备硅单晶的关键设备<sup>[16]</sup>。为了得到高品质硅单晶,通常通过调节加热器功率来有效控制炉内的热场温度,因此有必要建立精确的热场温度模型。本文针对拉晶过程中的引晶阶段,采用式(16)作为该阶段的热场温度模型,基于现场采集的大量数据,对模型参数进行离线辨识,进而得到热场温度模型。

$$\frac{T(s)}{P(s)} = \frac{K_1}{L_1 s^2 + L_2 s + 1} e^{-\tau s} \quad (16)$$

式中: $T(s)$ 和 $P(s)$ 分别为热场温度和加热器功率, $K_1$ 、 $L_1$ 、 $L_2$ 、 $\tau$ 为模型未知参数。

上面已证实了 LEALO 算法对包含多维参数的多峰函数的寻优能力,现将 LEALO 算法应用于热场温度模型的离线辨识<sup>[17]</sup>。设置种群个数为 30,最大迭代次数 1 000,辨识 20 次,所得参数如表 3 所示。模型离散化的采样周期为 0.1,真实数据采样间隔 $\Delta t = 6$  s。在加热器功率作用下,辨识所得最佳参数模型输出与硅单晶热场温度系统真实输出对比如图 4 所示。

图 4 表明,离线辨识所得最佳参数模型的输出较好地拟合了热场温度的真实数据,从而说明了 LEALO 算法具有较好的离线参数辨识能力。

表 3 参数辨识结果

Table 3 The parameters identification results

参数	最佳值	平均值
$K$	6.89	6.91
$L_1$	54.51	52.10
$L_2$	38.82	40.37
$\tau$	10.5	9.9

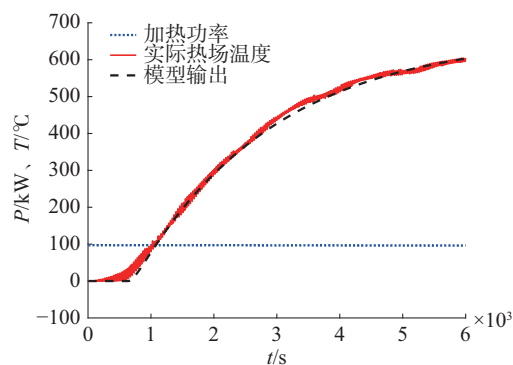


图 4 加热功率作用下热场真实温度和模型输出对比

Fig. 4 The contrast between true thermal field temperature and the output of model

## 4 结束语

作为一种新的寻优算法,蚁狮算法具有调节参数少、寻优精度高的特点。本文针对其缺点提出一种具有 Levy 变异和精英自适应竞争机制的 LEALO 算法。选择部分较差的个体进行 Levy 变异,保证了种群丰富度从而实现对求解区域进行充分探索,可以有效提高种群全局搜索能力;多个精英之间的并行竞争,削弱了单个精英对种群的误导,加快了寻优的收敛速度,精英竞争的自适应操作在保证寻优效率的同时,避免了较大的计算量。并与基本 ALO 算法及改进的 MALO 和 CIALO 算法进行对比,测试结果表明本文所提出的 LEALO 算法具有更好的寻优精度和收敛速度。最后将 LEALO 算法应用于硅单晶热场温度模型的参数辨识,仿真结果说明了该算法具有较好的优化能力。

## 参考文献:

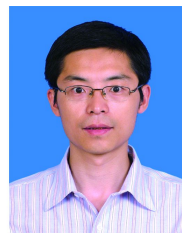
- [1] MIRJALILI S. The ant lion optimizer[J]. Advances in engineering software, 2015, 83: 80–98.
- [2] SAXENA P, KOTHARI A. Ant lion optimization algorithm to control side lobe level and null depths in linear antenna arrays[J]. AEU-International journal of electronics and communications, 2016, 70(9): 1339–1349.
- [3] HADIDIAN-MOGHADDAM M J, ARABI-NOWDEH S, BIGDELI M, et al. A multi-objective optimal Sizing and sit-

- ing of distributed generation using ant lion optimization technique[J]. Ain shams engineering journal, 2017, doi: 10.1016/j.asej.2017.03.001.
- [4] RAJU M, SAIKIA L C, SINHA N. Automatic generation control of a multi-area system using ant lion optimizer algorithm based PID plus second order derivative controller [J]. International journal of electrical power and energy systems, 2016, 80: 52–63.
- [5] RAJAN A, JEEVAN K, MALAKAR T. Weighted elitism based ant lion optimizer to solve optimum VAR planning problem[J]. Applied soft computing, 2017, 55: 352–370.
- [6] 赵世杰, 高雷阜, 于冬梅, 等. 带混沌侦查机制的蚁狮优化算法优化 SVM 参数[J]. 计算机科学与探索, 2016, 10(5): 722–731.
- ZHAO Shijie, GAO Leifu, YU Dongmei, et al. Ant lion optimizer with chaotic investigation mechanism for optimizing SVM parameters[J]. Journal of frontiers of computer science and technology, 2016, 10(5): 722–731.
- [7] REYNOLDS A. Liberating Lévy walk research from the shackles of optimal foraging[J]. Physics of life reviews, 2015, 14: 59–83.
- [8] 何莉, 王淼, 李博. 面向单目标优化的集成粒子群算法[J]. 重庆邮电大学学报: 自然科学版, 2017, 29(4): 527–534.
- HE Li, WANG Miao, LI Bo. Ensemble particle swarm optimizer for single objective optimization[J]. Journal of Chongqing university of posts and telecommunications: natural science edition, 2017, 29(4): 527–534.
- [9] REYNOLDS A M. Cooperative random Lévy flight searches and the flight patterns of honeybees[J]. Physics letters A, 2006, 354(5/6): 384–388.
- [10] 朱颢东, 孙振, 吴迪, 等. 基于改进蚁群算法的移动机器人路径规划[J]. 重庆邮电大学学报: 自然科学版, 2016, 28(6): 849–855.
- ZHU Haodong, SUN Zhen, WU Di, et al. Path planning for mobile robot based on improved ant colony algorithm[J]. Journal of Chongqing university of posts and telecommunications: natural science edition, 2016, 28(6): 849–855.
- [11] LEE C Y, YAO Xin. Evolutionary programming using mutations based on the levy probability distribution[J]. IEEE transactions on evolutionary computation, 2004, 8(1): 1–13.
- [12] MANTEGNA R N. Fast, accurate algorithm for numerical simulation of Lévy stable stochastic processes[J]. Physical review E, 1994, 49(5): 4677–4683.
- [13] JENSI R, JIJ G W. An enhanced particle swarm optimization with levy flight for global optimization[J]. Applied soft computing, 2016, 43: 248–261.
- [14] 江建. 精英自适应混合遗传算法及其实现[J]. 计算机工程与应用, 2009, 45(27): 34–35, 101.
- JIANG Jian. Elite adaptive hybrid genetic algorithm and its realization[J]. Computer engineering and applications, 2009, 45(27): 34–35, 101.
- [15] 周新宇, 吴志健, 王晖, 等. 一种精英反向学习的粒子群优化算法[J]. 电子学报, 2013, 41(8): 1647–1652.
- ZHOU Xinyu, WU Zhijian, WANG Hui, et al. Elite opposition-based particle swarm optimization[J]. Acta electronica sinica, 2013, 41(8): 1647–1652.
- [16] 刘丁. 直拉硅单晶生长过程建模与控制[M]. 北京: 科学出版社, 2015: 1–3, 46–57.
- LIU Ding. Modeling and controlling of crystal growth in the Czochralski process[M]. Beijing: Science Press, 2015: 1–3, 46–57.
- [17] 刘胜, 宋佳, 李高云. PSO 并行优化 LSSVR 非线性黑箱模型辨识[J]. 智能系统学报, 2013, 5(1): 51–56.
- LIU Sheng, SONG Jia, LI Gaoyun. Modeling a complex nonlinear system with particle swarm optimization and parallel-optimized least squares support vector regression[J]. CAAI transactions on intelligent systems, 2013, 5(1): 51–56.

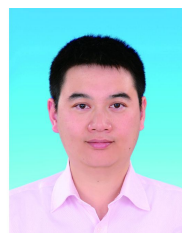
#### 作者简介:



景坤雷, 男, 1993 年生, 硕士研究生, 主要研究方向为非线性大时滞系统建模、参数辨识与控制。



赵小国, 男, 1978 年生, 讲师, 博士研究生, 主要研究方向为复杂系统的建模与控制。



张新雨, 男, 1985 年生, 讲师, 博士研究生, IEEE 会员, 主要研究方向为信号处理、自适应滤波、多目标优化和检测技术。先后发表学术论文 10 篇, 被 EI 检索 6 篇, 授权实用新型专利 2 项, 软件著作权 2 项, 申报发明专利 1 项。