

基于混合策略改进的麻雀搜索算法

胡树斌¹, 魏霖静²

(1. 甘肃农业大学理学院, 甘肃兰州 730070;
2. 甘肃农业大学信息科学技术学院, 甘肃兰州 730070)

摘要:为克服基本麻雀搜索算法(Sparrow Search Algorithm, SSA)存在的迭代过程中种群多样性减少、易陷入局部最优停滞、早熟收敛等不足,提出一种基于混合策略改进的麻雀搜索优化算法(EGSSA)。首先,迭代时利用精英反向学习机制在个体求解过程中计算精英反向解,提升算法寻优过程中种群丰富性和种群质量,为提高算法全局寻优能力以及收敛精度奠定基础;其次,将黄金正弦策略用以改进发现者位置更新方式,协调算法局部开发和全局搜索能力,并且提高算法的收敛能力;最后,在加入者的更新中采用了莱维飞行随机步长,以使新加入者所搜索的范围更为全面,一定程度避免算法陷入“早熟收敛”。选取12个基准测试函数进行仿真实验,分别与4种元启发式算法以及3种改进麻雀搜索算法进行比较,实验结果以及Wilcoxon秩和检验均表明,EGSSA算法具有更好的全局探索能力和稳定性以及较强跳出局部极值的能力。

关键词:麻雀搜索算法;精英反向学习;黄金正弦算法;莱维飞行;函数优化

中图分类号:TP18

文献标识码:A

文章编号:1673-629X(2023)04-0146-08

doi:10.3969/j.issn.1673-629X.2023.04.022

Improved Sparrow Search Algorithm Based on Hybrid Strategy

HU Shu-bin¹, WEI Lin-jing²

(1. School of Science, Gansu Agricultural University, Lanzhou 730070, China;

2. School of Information Science & Technology, Gansu Agricultural University, Lanzhou 730070, China)

Abstract: In order to overcome the shortcomings of the basic sparrow search algorithm (SSA) in the iterative process, such as population diversity reduction, local optimal stagnation and premature convergence, an improved sparrow search optimization algorithm (EGSSA) based on hybrid strategy is proposed. Firstly, the elite opposition-based learning mechanism is used to calculate the elite reverse solution in the individual solution process in the iteration, so as to improve the population richness and population quality in the optimization process of the algorithm, and lay the foundation for improving the global optimization ability and convergence accuracy of the algorithm. Secondly, the golden sine strategy is used to improve the location update method of the discoverer, coordinate the local development and global search ability of the algorithm, and improve the convergence ability of the algorithm. Finally, the random step size of Lévy flight is introduced in the update of the joiner's position, which makes the search direction of the joiner more comprehensive, and avoids the algorithm from falling into premature convergence. Twelve benchmark functions are selected for simulation experiments and compared with four other meta-heuristic algorithms and three other improved sparrow search algorithms, respectively. The experimental results and the Wilcoxon rank sum test both show that the EGSSA algorithm has better global exploration ability and stability, as well as a strong ability to jump out of local extremum.

Key words: sparrow search algorithm; elite opposition-based learning; golden-sine algorithm; Lévy flight; function optimization

0 引言

群智能优化算法(Intelligence Optimization Algorithm, IOA)灵感来自于自然界中物理现象或生物群体行为,可用于求解函数最优化问题,其方便易操作且

处理问题高效,为解决某些实际应用问题提供了新的思路。Xue等人^[1]受麻雀群体觅食活动启发,于2020年提出麻雀搜索算法(SSA)。试验表明,对比其他传统优化算法,它具有优化能力强、易于实现、调节参数

收稿日期:2022-05-31

修回日期:2022-09-30

基金项目:2020年甘肃农业大学研究生教育研究项目(2020-19);2021年度兰州市人才创新创业项目(2021-RC-47);2021年教育部产学研合作协同育人项目(202102326036)

作者简介:胡树斌(1996-),男,硕士研究生,CCF会员(K5635G),研究方向为应用统计大数据分析;通讯作者:魏霖静(1977-),女,博士,教授,CCF高级会员(57876S),研究方向为智能计算、生物信息学、农业信息化。

较少的优势。然而 SSA 算法同样存在不足,如迭代过程中种群丰富性降低、早熟收敛、易陷入局部最优停滞。为进一步提升原算法的性能,吕鑫等人^[2]借鉴鸟群算法思想,优化 SSA 算法的发现者和加入者迭代方式,并将改进算法应用于多阈值图像分割中,在减少分割时间的同时提升了分割精度。魏晓鸽等人^[3]采用精英反向学习策略改善初始种群,发现者位置迭代更新引入正余弦算法及动态权重因子,最后将改进算法用于火灾路径规划。付华等人^[4]首先通过立方混沌序列初始化麻雀种群,然后引入透镜成像反向策略提升初始解质量,借鉴鸡群算法优化跟随者位置更新方式,最后引入柯西-高斯变异策略对最优个体进行变异帮助算法脱离局部最优停滞状态。Yuan 等人^[5]使用重心反向学习策略改善初始种群质量,并用学习系数改善发现者位置变换方式提升算法全局寻优能力。Yan 等人^[6]将加入者追随发现者时进一步划分为全局与局部搜索,并通过观察超出边界个体的数量变化来验证可变螺旋因子与改进迭代搜寻策略的有效性。Jiang 等人^[7]引入随机游走策略,可有效帮助算法摆脱局部最优状态。Tang 等人^[8]借鉴遗传算法思想,将交叉和变异思想引入 SSA 算法中,保障种群多样性的同时可提升个体跳出局部最优的概率。

上述研究中学者们从各个方面对 SSA 算法的缺陷加以改善,提升算法性能,不过探索更多有效的改进策略提升算法收敛精度、全局寻优性能仍需深入研究。基于此,该文提出了一种基于混合策略改进的麻雀搜索算法。首先,利用精英反向学习机制改善种群迭代后期多样性降低的问题,为算法全局寻优奠定基础;然后,在发现者位置采用黄金正弦策略改进,提升算法开发能力;最后,加入者跟随发现者时融入莱维飞行步长,使搜索方向与范围多样化,避免算法“早熟收敛”。

1 基本麻雀搜索算法

麻雀搜索算法根据个体在觅食活动中的不同分工,将种群成员划分为发现者、加入者、警戒者。发现者与加入者身份可动态转换,但占群体比例不变。规模为 N 的麻雀种群可用矩阵表示为:

$$\begin{pmatrix} x_{1,1} & \cdots & x_{1,d} \\ \vdots & & \vdots \\ x_{N,1} & \cdots & x_{N,d} \end{pmatrix} \quad (1)$$

式中, N 为麻雀群体的规模, d 为待优化问题的空间维度。每只麻雀表示为 $x_i = [x_{i,1}, x_{i,2}, \dots, x_{i,d}]$, 适应度值为 $f(x_i)$, 全部麻雀的适应度值矩阵为 $F_x = [f(x_1), f(x_2), \dots, f(x_N)]$ 。

发现者为种群中位置较好的个体,占种群比例为 10% ~ 20%, 负责寻找食物。

其位置更新如下:

$$X_{i,j}^{t+1} = \begin{cases} X_{i,j}^t \cdot \exp\left(\frac{-i}{\alpha \cdot \text{Max} - \text{iter}}\right), R_2 < \text{ST} \\ X_{i,j}^t + Q \cdot L, R_2 \geq \text{ST} \end{cases} \quad (2)$$

式中, t 是当前迭代次数; $\text{Max} - \text{iter}$ 为最大迭代次数; $\alpha \in (0, 1)$ 为随机数,随机数 Q 服从正态分布; L 是元素取值均为 1 的 $1 \times d$ 矩阵。 $R_2 \in [0, 1]$ 为预警值; $\text{ST} \in [0.5, 1]$ 是安全阈值。

当 $R_2 < \text{ST}$ 时,表示麻雀觅食环境暂无危险,发现者可在边界范围内进行广泛搜索,否则说明察觉到捕食者的存在,需带领种群转移至安全区域。

加入者通过监视发现者的行为觅食,适应度值较高的在最佳位置的发现者附近搜索食物,适应度较低的则移动到空间其他位置。加入者位置更新如下:

$$X_{i,j}^{t+1} = \begin{cases} Q \cdot \exp\left(\frac{X_{\text{worst}}^t - X_{i,j}^t}{t^2}\right), i > \frac{N}{2} \\ X_p^{t+1} + |X_{i,j}^t - X_p^{t+1}| \mathbf{A}^+ \cdot L, \text{other} \end{cases} \quad (3)$$

式中, X_{worst}^t 为全局最差位置; X_p^{t+1} 是最佳发现者的位置; $1 \times d$ 的矩阵 \mathbf{A} 中元素由 -1 或 1 组成,并且满足 $\mathbf{A}^+ = \mathbf{A}^T (\mathbf{A} \mathbf{A}^T)^{-1}$ 。若 $i > N/2$, 表示第 i 只麻雀没有找到食物,处于能量较低状态,且因适应度值很差需飞往其他区域寻找食物。

警戒者数量占种群总数的 10% ~ 20%, 随机在发现者和加入者中选取,负责侦察警戒,初始位置更新公式为:

$$X_{i,j}^{t+1} = \begin{cases} X_{\text{best}}^t + \beta |X_{i,j}^t - X_{\text{best}}^t|, & f_i > f_{\text{best}} \\ X_{i,j}^t + K \left(\frac{|X_{i,j}^t - X_{\text{worst}}^t|}{(f_i - f_{\text{worst}}) + \varepsilon} \right), & f_i \neq f_{\text{best}} \end{cases} \quad (4)$$

式中, X_{best}^t 表示第 t 次迭代的最优位置,步长控制系数 β 服从标准正态分布,随机数 $K \in [-1, 1]$, ε 取最小常数,避免分母为 0,文中值为 $10\text{E}-50$, f_i 为个体适应度值, f_{worst} 为群体最差适应度值, f_{best} 为群体最佳适应度值。

2 改进麻雀搜索算法

该文混合三种有效策略改善 SSA 算法迭代时种群丰富性降低、早熟收敛、难以脱离局部最优现象,具体策略如下。

2.1 精英反向学习机制

反向学习机制^[9]靠近最优解的概率可提升 50%, 在可行解及其反向解中选择更优的个体作为新种群是其核心思想。精英反向学习 (Elite Opposition-Based Learning, EOBL) 则是在此基础上只针对种群精英个体构造反向种群。精英个体携带更多有效信息,相比较普通反向学习既提高了种群质量,又提升了搜索效率,有效地使种群避免寻优时掉入局部最优陷阱。

定义 1(精英反向解)。

假设 $X_{i,j}^E = (X_{i,1}^E, X_{i,2}^E, \dots, X_{i,d}^E)$, ($i = 1, 2, \dots, N$), ($j = 1, 2, \dots, d$) 为 d 维搜索空间的一个精英个体, 其反向解定义为 $\overline{X_{i,j}^E} = (\overline{X_{i,1}^E}, \overline{X_{i,2}^E}, \dots, \overline{X_{i,d}^E})$, 其中:

$$\overline{X_{i,j}^E} = c \cdot (\text{lb}_j + \text{ub}_j) - X_{i,j}^E \quad (5)$$

式中, $X_{i,j}^E \in [\text{lb}_j, \text{ub}_j]$, $c \in [0, 1]$ 为随机数, $\text{ub}_j = \max(X_{i,j}^E)$ 与 $\text{lb}_j = \min(X_{i,j}^E)$ 是动态搜索边界的上下界。搜索边界的动态变化可以保存之前的寻优经验, 提升搜索效率。对超过搜索边界的个体采用公式(6)重置:

$$\overline{X_{i,j}^E} = \text{rand}(\text{lb}_j + \text{ub}_j) \quad (6)$$

该文采用精英反向学习策略对初始种群以及迭代后期种群精英个体计算反向解时, 根据式(5)计算种群适应度排名前 $N/2$ 个个体的反向解, 将反向种群与原种群依据个体适应度排序, 挑选最优的 N 个精英个体组成下一代寻优种群。该策略可有效避免算法盲目搜索而降低求解效率, 为提升求解精度、加快收敛速度奠定基础。

2.2 黄金正弦机制

黄金正弦算法(Golden Sine Algorithm, Golden-SA)是 Tanyildizi 等人^[10]结合正弦函数原理在 2017 年提出, 其原理简单、易于操作、收敛性能好。最大特点是可遍历单位圆上全部正弦值, 其中黄金分割系数可帮助个体在迭代过程中极大程度缩小解空间, 并进行充分寻优, 提升了算法求解速度, 更有效平衡了算法全局“搜索”与“开发”。

Golden-SA 的核心是位置迭代方式, 数学描述如下:

$$X_i^{t+1} = X_i^t \times |\sin(r_1)| + r_2 \times \sin(r_1) \times |x_1 \times P_i^t - x_2 \times X_i^t| \quad (7)$$

式中, P_i^t 是当前全局最优位置, 随机数 $r_1 \in [0, 2\pi]$ 、 $r_2 \in [0, \pi]$ 分别决定个体位置迭代的移动距离与方向, 黄金分割系数 $x_1 = -\pi + (1 - \tau) \times 2\pi$ 、 $x_2 = -\pi + \tau \times 2\pi$ 通过黄金分割数所得, 黄金分割数 $\tau = (\sqrt{5} - 1)/2$ 。

观察 SSA 算法式(2)可知, 发现者在迭代初期就快速向全局最优靠近, 容易造成早熟收敛现象, 且相互之间缺乏足够的交流, 难以对优质解空间全面探索。针对此不足, 引入黄金正弦机制改善发现者探索方式, 改进后更新方式描述如下:

$$X_{i,j}^{t+1} = \begin{cases} X_{i,j}^t \times |\sin(r_1)| + r_2 \times \sin(r_1) \times |x_1 \times X_{i,\text{best}}^t - x_2 \times X_{i,j}^t|, & R_2 < \text{ST} \\ X_{i,j}^t + Q \cdot L, & R_2 \geq \text{ST} \end{cases} \quad (8)$$

2.3 莱维飞行机制

莱维飞行^[11]是对自然界中昆虫与鸟类觅食路径

随机游走过程的模拟, 其以大概率短距离搜索与小概率长距离搜索的方式交错移动, 产生服从重尾分布的随机步长^[12]。莱维飞行步长较大时可适当扩大搜索范围, 较小时可提升算法局部寻优能力, 其多样变化的特性使个体在空间内搜索更加全面。

观察 SSA 算法式(3)可知, 发现者找到食物时, 距离近的加入者会迅速靠近, 算法得以快速收敛, 但也因此造成短时间内麻雀在局部空间大量聚集, 种群多样性降低, 增大了算法“早熟收敛”的风险。因此, 该文在加入者位置更新中加入莱维飞行随机步长 step , 可以对与最优位置之间的空间全面搜索, 最大限度降低算法陷入局部最优陷阱的风险。莱维飞行产生的随机步长为:

$$\text{step} = \frac{\mu}{|v|^{\frac{1}{\beta_1}}} \quad (9)$$

其中, $\beta_1 = 1.5$, $\mu \sim N(0, \sigma_\mu^2)$, $v \sim N(0, \sigma_v^2)$ 。

$$\begin{cases} \sigma_v = 1 \\ \sigma_\mu = \frac{\Gamma(1 + \beta_1) \cdot \sin(\pi\beta_1/2)}{\beta_1 \cdot \Gamma[(1 + \beta_1)/2] \times 2^{(\beta_1-1)/2}} \end{cases} \quad (10)$$

式中, Γ 为标准伽玛函数; β_1 为固定步长, 一般取值为 1.5; σ_v 、 σ_μ 为正态分布中的标准差。式(3)引入莱维飞行随机步长 step 后改进为:

$$X_{i,j}^{t+1} = \begin{cases} Q \cdot \exp\left(\frac{X_{\text{worst}}^t - X_{i,j}^t}{t^2}\right), & i > \frac{N}{2} \\ X_p^{t+1} + |X_{i,j}^t - X_p^{t+1}| A^+ \cdot L \cdot \text{step}, & \text{other} \end{cases} \quad (11)$$

3 EGSSA 算法

3.1 EGSSA 算法步骤

(1) 参数设置。种群规模 N 、最大迭代次数 Max-iter 、发现者比例 PD 、警戒者比例 SD 、空间维度 d 、安全阈值 ST ;

(2) 初始化种群。对麻雀种群随机初始化, 根据目标函数计算麻雀个体的适应度值并排序;

(3) 计算精英反向解。选取种群排名前 $N/2$ 的麻雀个体根据式(5)求解反向种群, 根据式(6)对超过边界的个体进行重置, 计算反向种群适应度值;

(4) 从当前种群和反向种群中选取适应度值排名前 N 的个体组成新一代种群并依据个体适应度排序, 找到新一代种群中最差和最优适应度值: f_{worst} 、 f_{best} , 以及对应的麻雀个体位置 X_{worst} 、 X_{best} ;

(5) 根据公式(8)更新发现者位置;

(6) 根据公式(11)更新加入者位置;

(7) 根据公式(4)更新警戒者位置;

(8) 判断结束条件, 若迭代次数满足 $t > \text{Max-iter}$, 则终止操作, 输出全局最优位置及最优解, 否则

跳转步骤(3)继续执行。

3.2 EGSSA 算法伪代码

设置相关参数:种群规模 N 、最大迭代次数 Max-iter 、发现者比例 PD 、警戒者比例 SD 、空间维度 d 、安全阈值 ST ;

初始化种群,计算个体适应度值并排序

$t = 1$

While $t < \text{Max-iter}$:

for i in range($0, N/2$):

for j in range($0, d$):

根据公式(5)计算反向解

计算适应度值

检查是否超过边界,若超过,根据公式(6)重置

从当前种群和反向种群选取适应度值前 N 的个体组成新一代种群

对新种群按适应度值排序,记录最优、最差个体位置及适应度

随机取值 R_2 并判断与 ST 关系:

for i in range($0, \text{pNum}$):

根据公式(8)更新发现者位置

for i in range($\text{pNum}+1, N$):

根据公式(11)更新加入者位置

for i in range($0, N * \text{SD}$):

根据公式(4)更新警戒者位置

计算适应度值并更新最优个体适应度值及最优位置

返回最优位置及适应度值

if $t \% 1 == 0$:

print(在第 t 代,种群最优适应度值为: f_{best})

$t = t + 1$

3.3 EGSSA 时间复杂度分析

时间复杂度^[13]是反映算法性能的一个重要指标。假定麻雀种群规模为 N ,空间维度为 d ,求解目标函数所需的时间为 $f(d)$ 。根据文献[14],SSA 算法的时间复杂度为 $T = O(d + f(d))$ 。在 EGSSA 算法中,参数初始化的时间为 η_0 ,种群初始化的时间为 η_1 ,按式(5)生成反向种群所需时间为 η_2 ,则麻雀种群初始化阶段的时间复杂度为:

$$T_1 = O(\eta_0 + N(f(d) + d(\eta_1 + \eta_2))) = O(d + f(d)) \quad (12)$$

对上一代种群求精英反向解阶段,按式(5)更新麻雀精英个体所需的时间为 η_3 ,生成随机变量 $k \in [0, 1]$ 所需的时间为 η_4 ,对两个种群排序所需时间为 η_5 ,在这一阶段的时间复杂度为:

$$T_2 = O(N/2 \times ((\eta_3 + \eta_4)d + f(d)) + \eta_5) = O(d + f(d)) \quad (13)$$

麻雀种群中发现者数量为 $\text{PD} \times N$, PD 为发现者比例,每一维位置按式(8)进行更新的时间为 η_6 ,两个随机参数以及两个黄金分割系数生成时间均为 η_7 ,则该阶段时间复杂度为:

$$T_3 = O(\text{PD} \times N((\eta_6 + 4 \times \eta_7)d + f(d))) = O(d + f(d)) \quad (14)$$

麻雀种群中加入者数量为 $(1 - \text{PD}) \times N$, $(1 - \text{PD})$ 为加入者比例,每一维位置按式(11)进行更新的时间为 η_8 ,生成莱维飞行随机步长算子的时间为 η_9 ,则该阶段的时间复杂度为:

$$T_4 = O((1 - \text{PD}) \times N((\eta_8 + \eta_9)d + f(d))) = O(d + f(d)) \quad (15)$$

麻雀种群中警戒者数量为 $\text{SD} \times N$, SD 为警戒者比例,每一维位置按式(4)进行更新的时间为 η_{10} ,两个正态分布随机参数更新的时间为 η_{11} ,则这一阶段的时间复杂度为:

$$T_5 = O(\text{SD} \times N((\eta_{10} + 2 \times \eta_{11})d + f(d))) = O(d + f(d)) \quad (16)$$

最大迭代次数为 Max-iter ,综上,EGSSA 算法的时间复杂度为:

$$T = T_1 + \text{Max-iter}(T_2 + T_3 + T_4 + T_5) = O(d + f(d)) \quad (17)$$

综上可知,EGSSA 与标准 SSA 算法的时间复杂度在数量级上一致,该文针对标准 SSA 的不足所提出的改进策略没有增加计算负担。

4 仿真实验结果与分析

4.1 仿真实验环境

仿真实验环境为:操作系统 Windows10 (64 bit), CPU 为 Intel(R) Core(TM) i5-4200M,内存 12 GB,主频 2.50 GHz,仿真软件为 python3.9。

4.2 比较对象和参数设置

为验证改进 EGSSA 算法的优越性以及可行性,将提出的 EGSSA 算法分别与其他经典元启发式算法(鲸鱼优化算法(WOA)^[15]、粒子群算法(PSO)^[16]、灰狼算法(GWO)^[17]、飞蛾扑火算法(MFO)^[18])、三种其他文献所提改进算法(自适应变异麻雀算法(AMSSA)^[19]、自适应 t 分布与随机游走麻雀算法(ARSSA)^[20]、基于等级制度与布朗运动的混沌麻雀算法(CSSAHB)^[21]) 在基准函数上进行仿真对比,通过最优值、平均数以及标准差三个指标的比较来检验所提改进算法的求解精度与稳定性。所选的基准测试函数中 $F_1 - F_5$ 、 $F_6 - F_9$ 、 $F_{10} - F_{12}$ 依次是单峰、多峰、低维多峰函数,单峰函数用来验证算法的开发能力,多峰函数用来检验算法能否脱离局部最优。

所选基准测试函数见表1。设定种群规模 $N = 50$,

表 1 基准测试函数

Benchmark function	d	Range	f_{\min}
$F_1(x) = \sum_{i=1}^n x_i^2$	30	[-100,100]	0
$F_2(x) = \sum_{i=1}^n x_i^2 + \prod_{i=1}^n x_i $	30	[-10,10]	0
$F_3(x) = \sum_{i=1}^n (\sum_{j=1}^i x_j)^2$	30	[-100,100]	0
$F_4(x) = \max_i \{ x_i , 1 \leq i \leq n \}$	30	[-100,100]	0
$F_5(x) = \sum_{i=1}^n ix_i^4 + \text{random}[0,1]$	30	[-1.28,1.28]	0
$F_6(x) = \sum_{i=1}^n -x_i \sin(\sqrt{ x_i })$	30	[-500,500]	-418.982 9×dim
$F_7(x) = \sum_{i=1}^n [x_i^2 - 10\cos(2\pi x_i) + 10]$	30	[-5.12,5.12]	0
$F_8(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos(\frac{x_i}{\sqrt{i}}) + 1$	30	[-600,600]	0
$F_9(x) = -20\exp(-0.2\sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}) - \exp(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)) + 20 + e$	30	[-32,32]	0
$F_{10}(x) = -\sum_{i=1}^5 [(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	[0,10]	-10.153 2
$F_{11}(x) = -\sum_{i=1}^7 [(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	[0,10]	-10.402 8
$F_{12}(x) = -\sum_{i=1}^n [(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	[0,10]	-10.536 3

$F_1 - F_9$ 维度 $d = 30$, Max - iter 为 500。为保证公平,实验中各算法均独立运行 30 次。其余各参数设置见表 2。

表 2 各算法参数设置

算法	参数设置
SSA	PD=20%,SD=10%,ST=0.8
ESSA	PD=20%,SD=10%,ST=0.8
GSSA	PD=20%,SD=10%,ST=0.8
LSSA	PD=20%,SD=10%,ST=0.8
EGSSA	PD=20%,SD=10%,ST=0.8
AMSSA	PD=20%,SD=10%,ST=0.8,m=2
CSSAHB	PD=20%,SD=10%,ST=0.8,P=0.5
ARSSA	PD=20%,SD=10%,ST=0.8, $\eta = 1 - i / \text{iter}_{\max}$
GWO	$a_{\max} = 2, a_{\min} = 0$
MFO	——
PSO	$w = 0.9, c_1 = c_2 = 1.494 45$
WOA	$a \in [0,2]$ 并从 2 线性下降

4.3 EGSSA 与其他元启发式算法对比分析

为验证 EGSSA 的寻优性能,将提出的 EGSSA 算

法与传统元启发式算法:标准麻雀搜索算法(SSA)、鲸鱼优化算法(WOA)、粒子群算法(PSO)、灰狼算法(GWO)、飞蛾扑火算法(MFO)在 12 个测试函数上进行仿真实验,实验维度为 30,其余参数设置见表 2,独立运行 30 次的实验结果见表 3。

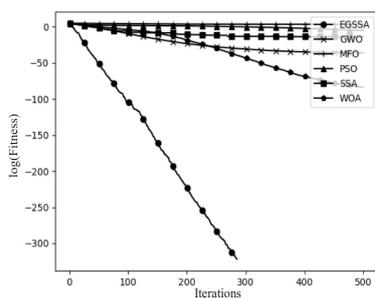
由表 3 的仿真实验结果可见,在所选 12 个基准测试函数上,EGSSA 算法寻优稳定性和求解精度均表现最优,在 F_4 函数上虽未找到最优值,但是平均值以及标准差是所选算法中最优的,算法收敛精度以及稳定性更能得到保证。SSA 算法在 $F_1, F_3, F_4, F_7, F_8, F_{10} \sim F_{12}$ 函数上虽能偶尔寻到最优解,但较小的平均值以及较大的标准差表明算法具有强烈的不稳定性,在低维固定维度函数 $F_{10} \sim F_{12}$ 上,其他群智能算法存在与标准 SSA 算法同样的问题,改进的 EGSSA 算法明显克服了这一不足,提高算法收敛精度的同时保证了算法求解的稳定性。综合对比实验结果的平均值、标准差可得出,EGSSA 的收敛精度更高,稳定性更好,明显优于其余 5 种算法。

为更直观地对比 EGSSA 与 GWO、MFO、PSO、WOA、SSA 的寻优速度和性能,图 1 给出在 12 个测试

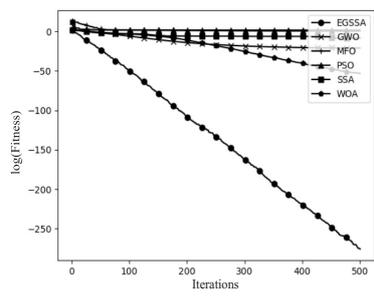
函数上独立运行 30 次的平均收敛曲线。图中横坐标 像作以 10 为底的对数处理。为迭代次数,为便于比较,纵坐标中函数值为正值的图

表 3 与其他群智能算法实验结果对比

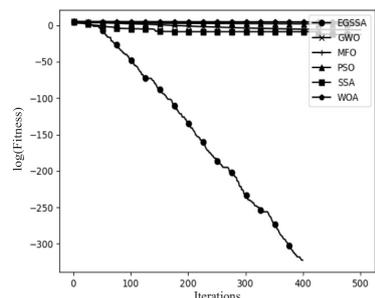
函数	GWO	MFO	PSO	WOA	SSA	EGSSA
最优值	3.93E-38	1.58E-02	2.83E-07	2.00E-94	0.00E+00	0.00E+00
F_1 平均值	7.69E-37	3.00E+03	1.31E-05	2.33E-84	1.77E-14	0.00E+00
标准差	8.88E-37	5.35E+03	1.88E-05	1.16E-83	9.69E-14	0.00E+00
最优值	1.42E-22	8.11E-02	7.00E-04	1.10E-60	5.78E-254	2.00E-304
F_2 平均值	8.40E-22	3.87E+01	3.68E+00	3.85E-54	3.59E-09	1.83E-276
标准差	5.28E-22	2.24E+01	5.56E+00	1.20E-53	1.93E-08	0.00E+00
最优值	1.01E-10	2.59E+03	1.49E+01	7.85E+03	0.00E+00	0.00E+00
F_3 平均值	6.25E-07	1.93E+04	4.63E+01	3.12E+04	1.68E-13	0.00E+00
标准差	1.76E-06	1.40E+04	1.90E+01	1.10E+04	8.10E-13	0.00E+00
最优值	2.74E-09	3.52E+01	4.74E-01	3.63E-01	0.00E+00	2.69E-319
F_4 平均值	4.74E-08	5.91E+01	7.93E-01	3.71E+01	4.47E-10	2.52E-290
标准差	5.83E-08	1.14E+01	1.76E-01	2.57E+01	2.13E-09	0.00E+00
最优值	6.12E-04	2.74E-02	3.95E-02	4.91E-05	1.18E-04	1.62E-05
F_5 平均值	1.62E-03	3.43E+00	2.85E+00	2.04E-03	6.70E-04	2.21E-04
标准差	6.52E-04	5.20E+00	5.06E+00	2.40E-03	4.73E-04	1.29E-04
最优值	-7.58E+03	-1.02E+04	-7.65E+03	-1.26E+04	-1.15E+04	-1.26E+04
F_6 平均值	-6.23E+03	-8.28E+03	-5.92E+03	-1.03E+04	-8.63E+03	-1.19E+04
标准差	1.09E+03	7.61E+02	1.11E+03	1.68E+03	1.36E+03	1.20E+03
最优值	1.14E-13	1.11E+02	3.78E+01	0.00E+00	0.00E+00	0.00E+00
F_7 平均值	5.51E+00	1.67E+02	8.31E+01	1.89E-15	2.67E-13	0.00E+00
标准差	6.77E+00	2.73E+01	2.41E+01	1.04E-14	1.28E-12	0.00E+00
最优值	0.00E+00	2.27E-02	6.77E-08	0.00E+00	0.00E+00	0.00E+00
F_8 平均值	4.12E-03	2.13E+01	7.39E-03	5.14E-03	8.13E-17	0.00E+00
标准差	7.28E-03	4.55E+01	8.02E-03	2.01E-02	4.45E-16	0.00E+00
最优值	2.89E-14	1.36E-01	4.26E-04	4.44E-16	4.44E-16	4.44E-16
F_9 平均值	3.85E-14	1.52E+01	2.54E-03	3.64E-15	1.12E-07	4.44E-16
标准差	4.19E-15	7.42E+00	2.18E-03	2.16E-15	3.93E-07	0.00E+00
最优值	-1.02E+01	-1.02E+01	-1.02E+01	-1.02E+01	-1.02E+01	-1.02E+01
F_{10} 平均值	-9.65E+00	-6.24E+00	-7.88E+00	-8.88E+00	-9.81E+00	-1.02E+01
标准差	1.54E+00	3.57E+00	2.68E+00	2.38E+00	1.29E+00	0.00E+00
最优值	-1.04E+01	-1.04E+01	-1.04E+01	-1.04E+01	-1.04E+01	-1.04E+01
F_{11} 平均值	-1.02E+01	-8.32E+00	-9.51E+00	-8.99E+00	-1.02E+01	-1.04E+01
标准差	9.63E-01	3.28E+00	2.38E+00	2.63E+00	9.70E-01	0.00E+00
最优值	-1.05E+01	-1.05E+01	-1.05E+01	-1.05E+01	-1.05E+01	-1.05E+01
F_{12} 平均值	-1.02E+01	-9.24E+00	-1.02E+01	-8.03E+00	-1.04E+01	-1.05E+01
标准差	1.37E+00	2.68E+00	1.37E+00	3.44E+00	9.87E-01	0.00E+00



(a)函数 F_1 收敛曲线



(b)函数 F_2 收敛曲线



(c)函数 F 收敛曲线

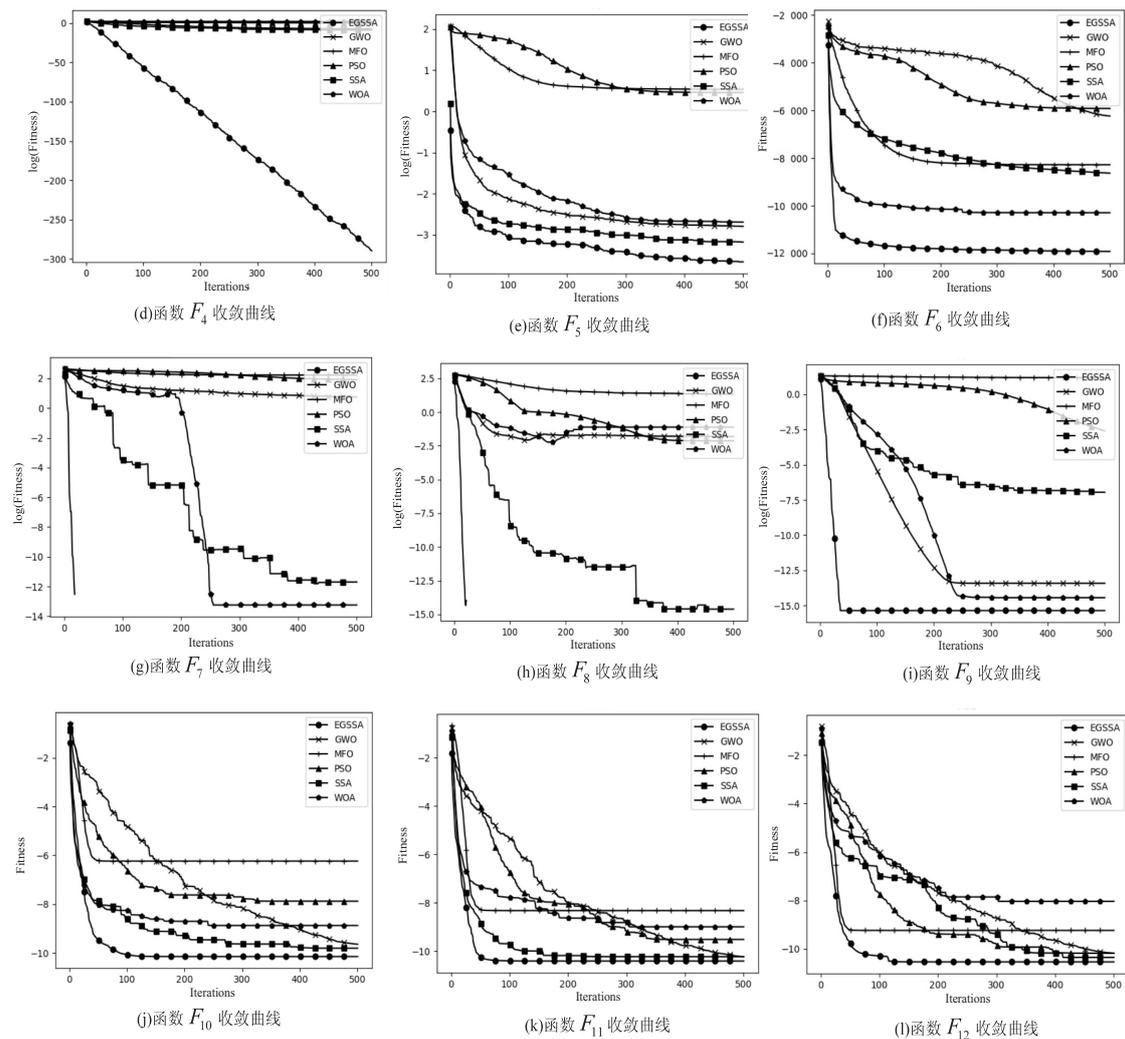


图1 不同群智能算法收敛曲线

由图1收敛曲线可看出,EGSSA算法的收敛曲线在12个基准测试函数上均表现更平滑,收敛速度更快。在单峰测试函数 F_1-F_4 上,EGSSA表现出远超于其他算法的收敛速度和求解精度(见图1(a)-(d));在 F_5 函数上虽未达到理论最有值,但结合收敛曲线图实验结果来看,EGSSA收敛速度在所有算法中最快的(见图1(e)),平均值与标准差更小;在 F_6-F_{12} 多峰测试函数上,EGSSA都能在迭代100次以内快速靠近全局最优,并未像其他算法一样陷入局部最优,与其他算法相比表现出明显的优越性(见图1(f)-(l))。

4.4 Wilcoxon 秩和检验

仅比较平均值和标准差并不能完全说明算法之间的差异性,为充分证明提出的EGSSA算法相比其他算法的优越性,本节采用统计方法Wilcoxon秩和检验来验证EGSSA算法与其他算法性能是否有显著差异。实验设定显著性水平为5%。当 $p < 0.05$ 时,拒绝原假设,判定EGSSA与其他算法具有显著差异;否则判定EGSSA与其他算法并无显著差异,寻优性能相当。

表4为在显著性水平5%下EGSSA与SSA、

GWO、MFO、PSO、WOA之间的Wilcoxon秩和检验结果。

表4 Wilcoxon秩和检验 p 值

函数	EGSSA - SSA	EGSSA - GWO	EGSSA - PSO	EGSSA - MFO	EGSSA - WOA
F_1	5.61E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06
F_2	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06
F_3	3.79E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06
F_4	1.92E-06	1.73E-06	1.73E-06	1.73E-06	1.73E-06
F_5	3.72E-05	1.73E-06	1.73E-06	1.73E-06	1.13E-05
F_6	2.13E-06	1.92E-06	1.73E-06	1.92E-06	1.38E-03
F_7	7.0E-02	1.69E-06	1.73E-06	1.73E-06	3.20E-01
F_8	3.20E-01	1.0E-02	1.73E-06	1.73E-06	1.80E-01
F_9	5.96E-05	1.15E-06	1.73E-06	1.73E-06	6.55E-06
F_{10}	1.73E-06	1.73E-06	6.32E-04	1.59E-04	1.73E-06
F_{11}	1.73E-06	1.73E-06	7.0E-02	1.0E-02	1.73E-06
F_{12}	1.73E-06	1.73E-06	1.10E-01	3.0E-02	1.73E-06

由表4数据可知,大部分 p 值远小于0.05,表明EGSSA在优化性能上优于所选传统智能优化算法。

4.5 EGSSA 与其他改进策略的 SSA 算法对比分析

为进一步验证所提算法具有优越性能,本节将 EGSSA 算法与文献[19]提出的自适应变异麻雀算法(AMSSA)、文献[20]提出的自适应 t 分布与随机游走麻雀算法(ARSSA)、文献[21]提出的基于等级制度与布朗运动的混沌麻雀算法(CSSAHB)进行对比分析。测试函数选取表1中 $F_1 - F_8$,其中 $F_1 - F_6$ 迭代次数为1000, F_7, F_8 两个函数迭代次数依旧为500。种群规模 $N = 50$,实验维度为30,各算法在测试函数上独立运行30次,其余参数设置见表2。为保证实验对比公平合理性,对比的三种改进算法实验结果与文献[20]数据结果相同,对比结果见表5。

表5 不同改进算法实验结果对比

函数	AMSSA	CSSAHB	ARSSA	EGSSA	
F_1	平均值	2.01E-317	0.00E+00	0.00E+00	0.00E+00
	标准差	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F_2	平均值	7.53E-150	7.60E-173	0.00E+00	0.00E+00
	标准差	4.12E-149	0.00E+00	0.00E+00	0.00E+00
F_3	平均值	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	标准差	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F_4	平均值	1.94E-147	1.02E-142	1.44E-173	0.00E+00
	标准差	1.07E-146	5.56E-142	0.00E+00	0.00E+00
F_5	平均值	4.30E-05	3.33E-05	5.80E-05	1.92E-04
	标准差	7.64E-05	4.26E-05	4.25E-06	1.57E-04
F_6	平均值	-1.01E+04	-1.08E+04	-1.11E+04	-1.14E+04
	标准差	2.60E+03	1.90E+03	1.71E+03	1.42E+03
F_7	平均值	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	标准差	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F_8	平均值	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	标准差	0.00E+00	0.00E+00	0.00E+00	0.00E+00

由表5可知,对于函数 F_3, F_7, F_8 所有改进算法均寻到理论最优值,且标准差为0;EGSSA 算法在函数 F_5 上各项指标略差于其他三种改进算法,在其余函数上,EGSSA 算法寻优精度及稳定性均优于 AMSSA 算法;对于函数 F_2 和 F_4 ,EGSSA 算法可以稳定收敛到理论最优,而其他算法仅 ARSSA 算法在 F_2 函数上寻到最优值;四种算法在函数 F_6 上均未稳定收敛到最优值,但是 EGSSA 算法的平均值和标准差两个指标均优于其他三种算法。综上,融合精英反向学习与黄金正弦的麻雀搜索算法寻优能力更强、稳定性更好,与其他策略改进算法相比同样具有一定的竞争优势。

5 结束语

针对标准 SSA 算法迭代后期种群多样性减少、早熟收敛等不足,提出一种基于混合策略改进的麻雀搜

索算法。在迭代过程中融入精英反向学习机制,丰富种群多样性,提高种群质量,提升算法搜索效率和收敛精度;在发现者搜索时采用黄金正弦策略,有效协调算法搜索能力和开发能力;最后在加入者靠近发现者时引入莱维飞行步长,增加个体位置变化多样性,有助于跳出局部最优,避免算法“早熟收敛”。选择单峰、多峰、固定维度多峰12个基准测试函数进行仿真实验,对比经典元启发式算法、其他改进算法寻优性能。研究表明,所提改进算法具有更好的收敛性能、全局寻优性能以及稳定性。

参考文献:

- [1] XUE J, SHEN B. A novel swarm intelligence optimization approach: Sparrow search algorithm [J]. Systems Science and Control Engineering, 2020, 8(1): 22-34.
- [2] 吕鑫, 慕晓冬, 张钧. 基于改进麻雀搜索算法的多阈值图像分割[J]. 系统工程与电子技术, 2021, 43(2): 318-327.
- [3] 魏晓鸽, 赵冠军. 基于改进麻雀搜索算法的疏散路径规划[J]. 消防科学与技术, 2022, 41(2): 221-226.
- [4] 付华, 刘昊. 多策略融合的改进麻雀搜索算法及其应用[J]. 控制与决策, 2022, 37(1): 87-96.
- [5] YUAN Jianhua, ZHAO Ziwei, LIU Yaping, et al. DMPPT control of photovoltaic microgrid based on improved sparrow search algorithm [J]. IEEE Access, 2021, 9: 16623-16629.
- [6] YAN S, YANG P, ZHU D, et al. Improved sparrow search algorithm based on iterative local search [J]. Computational Intelligence and Neuroscience, 2021, 2021: 6860503.
- [7] JIANG Z Y, GE J Q, XU Q Q, et al. Fast trajectory optimization for gliding reentry vehicle based on improved sparrow search algorithm [J]. Journal of Physics: Conference Series, 2021, 1986(1): 012114.
- [8] TANG Y Q, LI C H, LI S, et al. A fusion crossover mutation sparrow search algorithm [J]. Mathematical Problems in Engineering, 2021, 2021: 9952606.
- [9] TIZHOOSH H R. Opposition-based reinforcement learning [J]. Journal of Advanced Computational Intelligence and Intelligent Informatics, 2006, 10(4): 578-585.
- [10] TANYILDIZI E, DEMIR G. Golden sine algorithm: a novel math-inspired algorithm [J]. Advances in Electrical and Computer Engineering, 2017, 17(2): 71-78.
- [11] GHAEMI M, ZABIHINPOUR Z, ASGARI Y. Computer simulation study of the levy flight process [J]. Physica A: Statistical Mechanics and Its Applications, 2008, 388(8): 1509-1514.
- [12] 王庆喜, 郭晓波. 基于莱维飞行的粒子群优化算法 [J]. 计算机应用研究, 2016, 33(9): 2588-2591.
- [13] 周新, 邹海. 融合黄金正弦混合变异的自适应樽海鞘群算法 [J]. 计算机工程与应用, 2021, 57(12): 75-85.
- [14] 毛清华, 张强. 融合柯西变异和反向学习的改进麻雀算

(下转第160页)