



计算机科学与探索

Journal of Frontiers of Computer Science and Technology

ISSN 1673-9418, CN 11-5602/TP

## 《计算机科学与探索》网络首发论文

题目：融合学习机制的多混沌麻雀搜索算法  
作者：李光阳，潘家文，钱谦，殷继彬，伏云发，冯勇  
网络首发日期：2022-06-17  
引用格式：李光阳，潘家文，钱谦，殷继彬，伏云发，冯勇. 融合学习机制的多混沌麻雀搜索算法[J/OL]. 计算机科学与探索.  
<https://kns.cnki.net/kcms/detail/11.5602.tp.20220615.1924.002.html>



**网络首发：**在编辑部工作流程中，稿件从录用到出版要经历录用定稿、排版定稿、整期汇编定稿等阶段。录用定稿指内容已经确定，且通过同行评议、主编终审同意刊用的稿件。排版定稿指录用定稿按照期刊特定版式（包括网络呈现版式）排版后的稿件，可暂不确定出版年、卷、期和页码。整期汇编定稿指出版年、卷、期、页码均已确定的印刷或数字出版的整期汇编稿件。录用定稿网络首发稿件内容必须符合《出版管理条例》和《期刊出版管理规定》的有关规定；学术研究成果具有创新性、科学性和先进性，符合编辑部对刊文的录用要求，不存在学术不端行为及其他侵权行为；稿件内容应基本符合国家有关书刊编辑、出版的技术标准，正确使用和统一规范语言文字、符号、数字、外文字母、法定计量单位及地图标注等。为确保录用定稿网络首发的严肃性，录用定稿一经发布，不得修改论文题目、作者、机构名称和学术内容，只可基于编辑规范进行少量文字的修改。

**出版确认：**纸质期刊编辑部通过与《中国学术期刊（光盘版）》电子杂志社有限公司签约，在《中国学术期刊（网络版）》出版传播平台上创办与纸质期刊内容一致的网络版，以单篇或整期出版形式，在印刷出版之前刊发论文的录用定稿、排版定稿、整期汇编定稿。因为《中国学术期刊（网络版）》是国家新闻出版广电总局批准的网络连续型出版物（ISSN 2096-4188，CN 11-6037/Z），所以签约期刊的网络版上网络首发论文视为正式出版。

# 融合学习机制的多混沌麻雀搜索算法

李光阳<sup>1,2</sup>, 潘家文<sup>1,3</sup>, 钱谦<sup>1,2+</sup>, 殷继彬<sup>1,2</sup>, 伏云发<sup>1</sup>, 冯勇<sup>1,2</sup>

1.昆明理工大学信息工程与自动化学院, 昆明 650500

2.昆明理工大学云南省计算机技术应用重点实验室, 昆明 650500

3.中国农业大学信息与电气工程学院, 北京 100083

+ 通讯作者 E-mail:Qianqian\_yn@126.com

**摘要：**针对麻雀搜索算法(sparrow search algorithm, SSA)易受初始解的影响陷入局部极值、迭代后期收敛速度慢等缺陷, 提出了一种融合学习机制的多混沌麻雀搜索算法(Multi-chaotic Sparrow search algorithm based on learning mechanism, MMCSSA)。该算法首先引入重心反向学习策略(centroid opposition-based learning, COBL)生成精英种群增强对多源优质搜索区域的勘探能力, 提升算法的局部极值逃逸能力和收敛性能;其次, 提出一种动态调整的黄金正弦领导策略并嵌入 SSA 中以改善发现者的搜索方式, 增强算法的全局搜索能力;然后, 提出一种基于学习机制的多混沌映射策略, 该机制利用多混沌多扰动模式的特性, 通过动态调用不同混沌映射赋予算法不同类别的扰动特征, 混沌扰动失败时, 引入高斯变异策略对当前解进行深度开发, 两种策略协同作用, 相互促进, 极大增强了算法逃逸局部最优的能力。最后, 将所提算法应用于 12 个不同特征的基准函数进行实验, 结果表明与其他算法相比, MMCSSA 在收敛精度、寻优速度和鲁棒性方面有更好的表现。

**关键词：**麻雀搜索算法;黄金正弦算法;高斯变异;多混沌学习机制;重心反向学习策略

**文献标志码：**A **中图分类号：**TP391

## Multi-chaotic sparrow search algorithm based on learning mechanism

LI Guangyang<sup>1,2</sup>, PAN Jiawen<sup>1,3</sup>, QIAN Qian<sup>1,2+</sup>, YIN Jibin<sup>1,2</sup>, FU Yunfa<sup>1</sup>, FENG Yong<sup>1,2</sup>

1.Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China

2.Yunnan Key Laboratory of Computer Technology Applications, Kunming University of Science and Technology, Kunming 650500, China

3.Faculty of Information and Electrical Engineering, China Agricultural University, Beijing 100083, China

**Abstract:** To solve the shortcomings of sparrow search algorithm (SSA), such as falling into local extremum easily influenced by initial solution and slow convergence in late iteration, a multi-chaotic Sparrow search algorithm (MMCSSA) based on learning mechanism was proposed. Firstly, the centroid opposition-based learning strategy (COBL) is introduced to generate elite population to enhance the exploration of multi-source high-quality search areas, and then the local extreme escape ability and convergence performance of the algorithm are improved. Secondly, a scaled golden sine algorithm is proposed and embedded in SSA to improve the guidance search mode and enhance the global search ability of the algorithm. Thirdly, a multi-chaos mapping strategy based on learning mechanism is proposed, which utilizes the characteristics of multi-chaos and multi-disturbance, and enforces the algorithm with different disturbance features by dynamically calling different chaotic maps. When chaotic perturbation fails, gaussian mutation strategy is introduced to deeply develop the current solution. The two strategies cooperate and promote each other, greatly enhancing the ability of the algorithm to escape the local optimal. Finally, the proposed algorithm is applied to 12 benchmark functions with different characteristics, and the results show that compared with other algorithms, MMCSSA has better performance in convergence accuracy, optimization speed and robustness.

**Key words:** Sparrow search algorithm; Golden sine algorithm; Gauss mutation; Multiple chaotic learning mechanisms; Center of gravity reverse learning strategy

麻雀搜索算法(sparrow search algorithm, SSA)<sup>[1]</sup>是薛建凯等人在 2020 年提出的一种新型群智能算法, 该算法受麻雀捕食行为的启发, 将群体分为发现者、

跟随者和警戒者, 并根据麻雀遭遇天敌的应对策略来构建算法的数学模型。目前, SSA 已广泛应用于路径规划<sup>[2]</sup>、任务调度<sup>[3]</sup>, 网络覆盖<sup>[4]</sup>等领域。

**基金项目：**国家自然科学基金(32060193); 云南省科技厅基础研究专项(202101AT070082); 云南省计算机技术应用重点实验室开放基金资助。

This work was supported by the the National Natural Science Foundation of China (32060193), the Basic Research Project of Science and Technology Department of Yunnan Province (202101AT070082) and the the Open Foundation of Yunnan Provincial Key Laboratory of Computer Technology Application.

SSA 因收敛速度快、扩展性与鲁棒性强等特点, 受到不少学者的关注与研究<sup>[5]</sup>。但 SSA 仍存在一些缺陷, 例如收敛精度依赖于初始解的质量、搜索过程中种群多样性衰减过快导致其易陷入局部极值、发现者易趋近于原点影响算法的收敛能力等。目前已有许多解决这类问题的方法, 如混合扰动机制<sup>[6]</sup>、将麻雀搜索算法与其他算法结合<sup>[7]</sup>等, 这些方法虽然在一定程度上提升了 SSA 的性能, 但仍存在进一步提高的可能性。现有的面向 SSA 性能改进的方法大致可以分为以下几类:

1) 种群初始化: 尹德鑫等人<sup>[8]</sup>在 SSA 中引入反向学习机制(opposition-based learning, OBL)增加了初始种群的多样性, 但 OBL 在计算反向点时仅仅考虑了搜索空间的边界信息, 并没有充分考虑种群作为一个寻优整体所携带的有利信息, 导致其优化效率仍然较低;张伟康等<sup>[9]</sup>利用 Circle 混沌映射产生初始种群, 在一定程度上增强了算法的全局探索能力, 但 Circle 混沌序列分布不够均匀, 且在[0.2,0.6]之间的取值较为密集, 这种现象限制了初始种群的多样性。为克服上述缺陷, 本文借鉴重心反向学习<sup>[10]</sup>的思想提出了一种重心反向学习(centroid opposition-based learning, COBL)策略, 该策略根据种群的重心来计算反向解, 有效提升了初始解的多样性。

2) 个体位置更新: 付华等人<sup>[11]</sup>借鉴鸡群优化算法(chicken swarm optimization algorithm, CSOA)改良了 SSA 中跟随者的位置更新方式, 提高了解的精度, 但该方法仍存在着缺陷, CSOA 中的个体因移动幅度过大而频繁越界, 影响了算法的收敛速度;温泽宇等人<sup>[12]</sup>在发现者的位置更新公式中引入了多项式变异因子, 增强了算法逃逸局部最优的能力, 但仍未从根本上解决发现者趋于原点的问题。因此, 本文借鉴黄金正弦算法(Golden sine algorithm, Gold-SA)<sup>[13]</sup>提出一种领导策略, 该策略在发现者位置更新公式中加入最优个体来引导其余个体的搜索方向, 增强了算法的全局搜索能力。此外, 为平衡算法的勘探能力与开发能力, 本文在领导策略中加入缩放因子, 进一步提升了算法的性能。

3) 群体变异扰动: 吕鑫等人<sup>[14]</sup>利用 tent 混沌映射来扰动算法的种群, 增强了算法摆脱局部最优的能力, 但该方法受限于其单一的扰动模式使算法的性能提升有限;李爱莲等人<sup>[15]</sup>使用柯西变异对跟随者进行扰动, 一定程度上强化了算法的全局寻优能力, 但柯西变异本质上属于邻域扰动, 当最优解不在当前个体附近时, 其扰动效果较为有限。基于上述不足, 本文提出一种基于学习机制的多混沌映射策略, 该策略以轮盘赌的形式为算法动态地选择合适的混沌映射, 大大提高了个体跳出局部最优的成功率。此外, 由于混沌扰动属于全局扰动, 其在算法局部区域扰动的能力较差, 为了弥补该缺陷, 在混沌映射策略的基础上引

入了高斯变异策略。

综上所述, 本文提出一种融合学习机制的多混沌麻雀搜索算法(Multi-chaotic Sparrow search algorithm based on learning mechanism, MMCSSA), 算法中的四项改进策略协同作用, 相互促进提升了算法的性能。为验证 MMCSSA 的有效性, 采用 12 个基准测试函数进行测试, 同时进行了 Friedman 检验与 Nemenyi 统计实验。实验结果验证了 MMCSSA 的优越性和可行性。

## 1 麻雀搜索算法

在 SSA 中, 存在三种主要的群体: 发现者、跟随者和警戒者。其中, 适应度较高的个体被称为发现者, 其作用是带领跟随者寻找食物, 相较于其他个体, 发现者有更广泛的搜索范围, 位置更新公式如下:

$$X_{i,j}^{t+1} = \begin{cases} X_{i,j}^t \cdot \exp(-\frac{i}{\alpha \cdot iter})R2 < ST \\ X_{i,j}^t + Q \cdot LR2 \geq ST \end{cases} \quad (1)$$

式(1)中,  $i$ 表示第 $i$ 只麻雀,  $i = 1, 2, 3 \dots n$ ;  $j$ 表示优化问题的第 $j$ 个维度,  $j = 1, 2, 3 \dots D$ ;  $t$ 表示算法第 $t$ 次迭代;  $iter$ 表示最大迭代次数;  $\alpha \in (0, 1]$ , 是一个随机数;  $R2 \in [0, 1]$ 代表预警值;  $ST \in [0.5, 1]$ 表示安全值;  $Q$ 是服从正态分布的随机数;  $L$ 表示 $1 \times D$ 的矩阵; 当 $R2 < ST$ 时, 说明种群处于安全环境, 此时发现者应扩大搜索范围; 当 $R2 \geq ST$ 时, 表明种群的周围出现了天敌, 为了保证安全, 整个种群飞行到安全区域进行觅食。

在觅食过程中, 跟随者会时刻监视发现者, 一旦看到发现者寻找到了食物, 跟随者会过去争抢食物, 若争抢失败, 则飞行到其他位置继续觅食。跟随者的位置更新公式如下:

$$X_{i,j}^{t+1} = \begin{cases} Q \cdot \exp(-\frac{X_{worst}^t - X_{i,j}^t}{i^2}) & i > \frac{n}{2} \\ X_p^{t+1} + |X_{i,j}^t - X_p^{t+1}| \cdot A^+ \cdot Lother & \end{cases} \quad (2)$$

式(2)中,  $X_{worst}$ 表示当前种群中的最差位置;  $X_p$ 是目前为止发现者探明的最优位置;  $A$ 表示为一个 $1 \times D$ 的矩阵, 其中每个元素随机为 1 或-1,  $A^+ = A^T(AA^T)^{-1}$ ; 当 $i > \frac{n}{2}$ 时, 表示这部分跟随者没有抢到食物, 需要移动到其他位置进行觅食; 当 $i \leq \frac{n}{2}$ 时, 说明跟随者与发现者争夺食物。

种群觅食时, 部分麻雀被选作警戒者负责侦查预警, 当天敌靠近时, 它们将会放弃当前的食物并向其他麻雀靠近以躲避危险。随机选出的警戒者通常在种群中占 10%或 20%左右, 警戒者的位置更新公式如下:

$$X_{i,j}^{t+1} = \begin{cases} X_{best}^t + \beta \cdot |X_{i,j}^t - X_{best}^t| f_i > f_g \\ X_{i,j}^t + k \cdot (\frac{|X_{i,j}^t - X_{worst}^t|}{f_i - f_w + \epsilon}) f_i = f_g \end{cases} \quad (3)$$

式(3)中 $X_{best}$ 表示当前种群中的最优位置;  $\beta$ 是标准正态分布随机数, 用来控制移动步长;  $k$ 是 $[-1, 1]$ 内的

一个随机数,用于控制麻雀的移动方向和距离; $f_i$ 表示第*i*个麻雀的适应度值; $f_g$ 和 $f_w$ 分别表示当前全局最优位置和全局最差位置的适应度值;为了避免分母为0,引入最小的常数 $\varepsilon$ ;当 $f_i > f_g$ 时,麻雀易受天敌攻击,当 $f_i = f_g$ 时,麻雀意识到了危险并向其他麻雀靠近。

## 2MMCSSA 的基本思想

### 2.1 重心反向学习策略

有效控制 SSA 初始种群的质量是提高算法性能的一个重要途径。初始种群的质量主要取决于其搜索范围和起始位置两个因素,若初始种群的搜索范围过小,会影响算法的勘探能力;若起始位置靠近全局最优解,种群能在更优质的解空间内挖掘有效信息。标准 SSA 采用随机初始化方法产生种群,该方法难以满足上述两个要求。为解决这个问题,算法考虑引入反向学习策略来产生初始种群,使用反向学习策略能够在更广阔的范围内找到优质解,进而有效提升初始种群的质量。但该方法仍存在一些缺陷,反向学习的原理是利用搜索空间的边界信息计算个体的反向解,但在面对含有“对称山峰”的优化问题时<sup>[16]</sup>会“失效”(当前解的适应度与其反向解的适应度相同)。为克服这个缺陷,本文提出一种重心反向学习策略来产生初始种群,其基本思想是根据种群的重心来计算反向解。重心反向学习策略初始化种群的方式如下,其中,离散均匀体 $C_j$ 表示为种群重心, $\bar{X}_i$ 是 $X_i$ 的重心反向点。

$$\bar{X}_i = 2 \cdot C - X_i \quad (4)$$

$$C_j = \frac{\sum_{i=1}^n X_{ij}}{n} \quad j = 1, 2, 3 \dots D \quad (5)$$

### 2.2 动态调整的黄金正弦领导策略

影响 SSA 性能的另一缺陷是发现者搜索范围较小且易趋于原点,这导致了算法易陷入局部最优。当 $R2 < ST$ 的条件满足时,SSA 根据式(1)更新发现者的位置,式(1)中的 $\exp(-\frac{i}{\alpha \cdot iter})$ 决定了发现者的勘探能力,假设其计算所得值为 $p$ ,则 $p$ 的取值变化如图 1 所示。根据*i*取值的不同, $p$ 的值常落于 $[0.9, 1]$ 内,这会导致发现者的移动范围过小,影响算法的全局探索能力;其次, $p$ 的取值总是小于 1,随迭代次数的增加,发现者的所有元素都不断地向 0 靠近,由于发现者起着引导种群的作用,上述现象将导致种群中所有个体都趋于原点,这使得 SSA 求解最优解位置非原点的问题时,算法的搜索速度和收敛精度都受到影响。当条件 $R2 \geq ST$ 被满足时,算法会在个体的每个维度都赋予相同的正态随机数以实现位置扰动,但该方法在定义域范围较大的目标问题上所达到的效果有限。为解决上述问题,本文采用黄金正弦领导策略更新发现者的位置:

$$X_i^{t+1} = X_i^t \cdot |\sin(r_1)| + r_2 \cdot \sin(r_1) \cdot |x_1 \cdot P^t - x_2 \cdot X_i^t| \quad (6)$$

式(6)中, $X_i$ 表示种群的第*i*个个体; $t$ 是迭代次数; $P$

为食物(全局最优个体); $r_1$ 和 $r_2$ 都是随机数, $r_1$ 的随机范围是 $[0, 2\pi]$ , $r_1$ 决定了下一次迭代中当前个体的移动距离, $r_2$ 的随机范围是 $[0, \pi]$ , $r_2$ 决定了下一次迭代中当前个体的位置更新方向; $x_1$ 和 $x_2$ 是引入黄金分割系数得到的参数,这些参数共同起到缩放算法搜索空间的作用,有利于加快算法的收敛速度,其中 $x_1 = (-\pi) + 2\pi \cdot (1 - \tau)$ , $x_2 = (-\pi) + \tau \cdot 2\pi$ , $\tau = \frac{\sqrt{5}-1}{2}$ 。

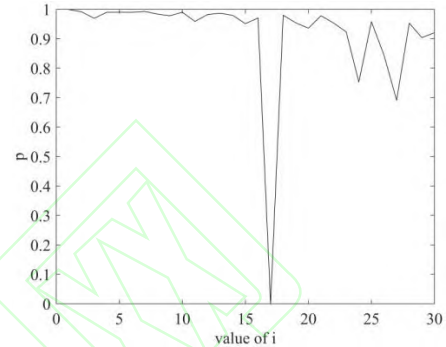


Fig.1 Value distribution of the parameter p  
图 1 参数 p 的数值分布

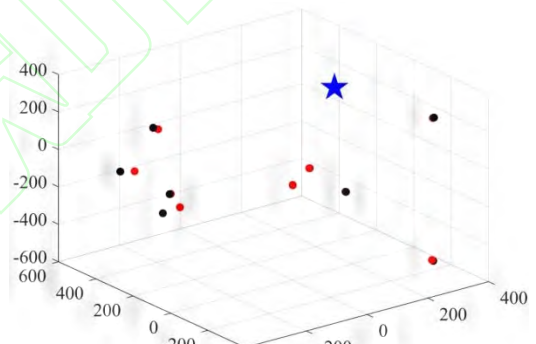


Fig.2 Original finder movement  
图 2 原始发现者移动

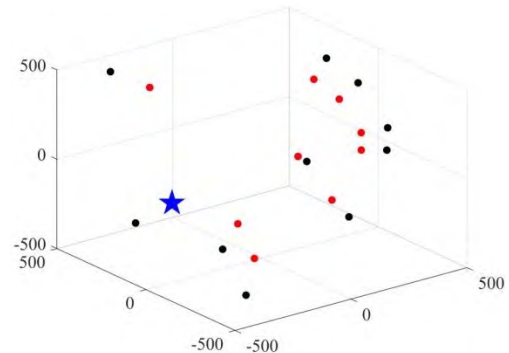


Fig.3 Improved Finder movement  
图 3 改进后的发现者移动

图 2 和图 3 分别表示某次实验中原始的发现者与改进后的发现者的移动变化。图中黑色圆点和红色圆点分别表示个体移动前与移动后的位置,蓝色五角星表示种群当前最优解的位置。由图 2 可知,原始的发现者更新前后的位置有大量重合点,未重合情况下的移动距离也偏小,且所有点都朝空间中的原点移动,

搜索方向较为单一，影响了算法的勘探能力;由图 3 可知，本文改进后的发现者在移动时会朝着多个方向移动，且移动距离较大，算法的全局搜索能力得到增强，同时，种群中其他个体也会朝最优解所在的方向移动，进一步增强了算法的收敛能力。由分析可知，本文引入的黄金正弦领导策略改善了发现者的移动路径，避免了种群易趋于原点的缺陷。

为了进一步调节算法在勘探能力和开发能力之间的平衡，本文在发现者的位置更新公式中引入缩放因子 $w$ 。

$$\mathbf{X}_i^{t+1} = \mathbf{X}_i^t \cdot |\sin(r_1)| + r_2 \cdot \sin(r_1) \cdot |w \cdot x_1 \cdot \mathbf{P}^t - x_2 \cdot \mathbf{X}_i^t| \quad (7)$$

$$w = \frac{\sin(\frac{i\pi}{M} - \frac{\pi}{2}) + 1}{2} \quad (8)$$

上式中 $i$ 为迭代次数， $M$ 代表最大迭代次数。缩放因子 $w$ 与全局最优解 $P$ 共同决定了最优解引导种群的能力。图 4 表示动态因子 $w$ 随迭代次数变化的函数曲线（实线表示），相比于线性变换(虚线表示)， $w$ 在迭代初期上升缓慢，全局最优解的引导力较小，有效避免了前期因聚集而导致的种群多样性减少，保证了算法前期的搜索能力;随着算法进入迭代后期， $w$ 值快速增加，使最优解对种群的引导力显著增强，种群更迅速聚集在最优解周围进行开发，增强了算法的收敛能力。由上述分析可知， $w$ 值的变化决定了全局最优解对种群的引导能力，通过分阶段的调节 $w$ 值，避免了算法前期搜索能力较弱，后期收敛速度较慢的情况，更好地平衡了算法在不同时期勘探能力和开发能力之间的平衡。

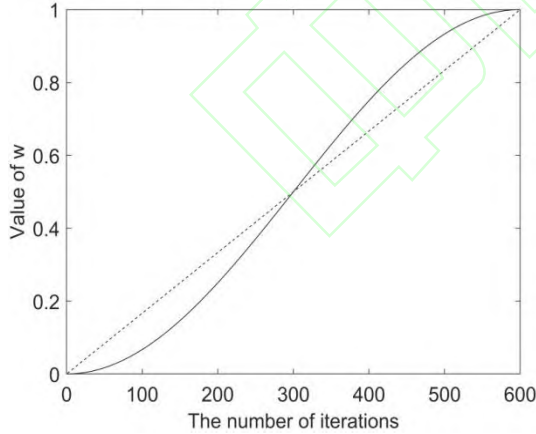


Fig.4 The scaling factor  $w$  varies with iteration  
图 4 缩放因子 $w$ 随迭代变化曲线

### 2.3 学习选择机制的多混沌映射策略

SSA 在搜索后期因种群多样性急剧减少易陷入局部最优，使得 SSA 收敛精度降低。因此，需要在计算过程中对个体进行持续扰动来增加算法迭代后期的多样性。利用混沌映射进行扰动是改善上述缺陷的一个有效方法，但已有研究大多使用单一的混沌映射对个体进行扰动，扰动效果一般。为提升扰动效率，本文

在前人研究的基础上<sup>[17,18]</sup>提出了一种基于学习机制的多混沌映射策略。

#### 2.3.1 多混沌映射模型

通过对常见的混沌映射进行比较。本文选出 7 个具有代表性的混沌映射:概率分布均匀的 Kent 映射，概率分布非均匀的 Logistic 映射、Sine 映射、ICMIC 映射、Circle 映射、Chebyshev 映射、Gauss 映射，用于该多混沌映射模型。每种混沌映射都有不同的分布特性，例如 Circle 映射的概率分布呈现山峰形态，即中间有较高的概率密度值，而两边的密度值较低;ICMIC 映射的概率分布是一个山谷形态，中间的概率密度值较低，两边较高。每次迭代后期选择混沌映射对种群中所有个体进行扰动，其计算公式和用到的混沌映射如下:

$$\mathbf{X}_i^t = \mathbf{X}_i^t + r \cdot (\mathbf{U} - \mathbf{L}) \cdot (z^t - 0.5) \quad (9)$$

式(9)中 $\mathbf{U}$ 和 $\mathbf{L}$ 分别是搜索空间上边界和下边界; $r = r \times 0.988$ 为混沌搜索半径变化公式，其搜索半径随迭代次数的增加而减小，使得算法在迭代后期有更快的收敛速度; $z^t$ 表示在第 $t$ 代时所使用的混沌变量。个体经混沌扰动再使用贪心策略保留较优个体，其数学表达式为:

$$\mathbf{X}_i^{t+1} = \begin{cases} \mathbf{X}_i^t, f(\mathbf{X}_i^t) < f(\mathbf{X}_i^t) \\ \mathbf{X}_i^t, f(\mathbf{X}_i^t) \geq f(\mathbf{X}_i^t) \end{cases} \quad (10)$$

#### 2.3.2 学习选择机制

在某些研究中<sup>[19,20]</sup>，机器学习方法被用来分析历史数据(如种群进化信息和问题特征信息)，以便指导算法。借鉴该思想，本文使用学习周期的历史信息来决定混沌转盘的概率，这种概率是种群当前搜索特征的本质表达。如图 5 所示，混沌映射所对应的概率越大，表明该混沌映射对种群的影响越大，在下一周期有更高的概率扰动成功(寻得更优解)。用 $S_j^t$ 表示第 $t$ 次迭代时第 $j$ 个混沌映射扰动成功的次数，计算公式如下:

$$S_j^{t+1} = \begin{cases} S_j^t + 1, f(\mathbf{X}_i^t) < f(\mathbf{X}_i^t) \\ S_j^t, f(\mathbf{X}_i^t) \geq f(\mathbf{X}_i^t) \end{cases} \quad (11)$$

每过一个学习周期，成功次数 $S_j^t$ 需要重置为 0。第一个学习周期中默认每个混沌映射以均匀概率被选择，之后的每个学习周期混沌映射的选择概率由以下公式计算:

$$P_j^c = \frac{S_j^c}{n \cdot t_j} \quad (12)$$

式(12)中 $c$ 指第 $c$ 个学习周期， $n$ 表示种群中的个体数量， $t_j$ 是第 $j$ 个混沌映射在一个学习周期内被选中的次数。MMCSSA 在每个学习周期的间隔都会通过式(12)更新概率轮盘并为下一学习周期选择混沌映射。

五个映射的初始概率轮盘

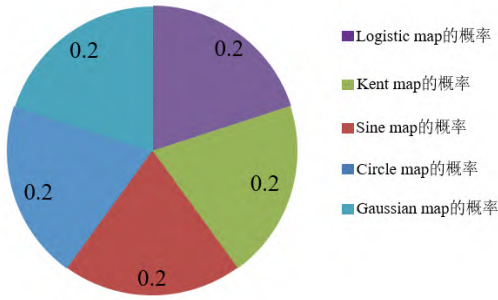


Fig.5 Initial probability roulette

图 5 初始概率轮盘

## 2.4 引入高斯变异策略

多混沌映射策略的作用域是整个搜索空间，扰动失败意味着搜索空间其他区域有更优解的可能性较小；由高斯分布的概率分布可知，高斯变异策略是在个体邻域内进行深度开发，有利于提高解的收敛精度。为了进一步提升算法的性能，对个体扰动时，若混沌扰动失败就使用高斯变异进行二次扰动。高斯分布的概率密度函数和高斯变异策略的数学模型分别如下所示：

$$f(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} \quad (13)$$

$$X_i^{t+1} = X_i^t \cdot Gauss(1,0.1) \quad (14)$$

## 2.5 MMCSSA 实现流程

MMCSSA 的具体实现流程如图 6 所示：

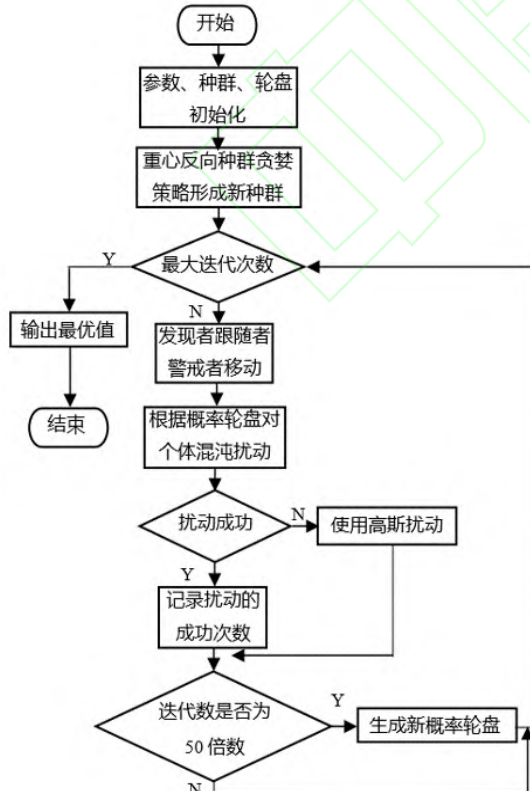


Fig.6 Flow chart of improved algorithm

图 6 改进算法流程图

Step1:初始化种群 $X$ 、各项参数，设置混沌概率轮

盘；

Step2:初始种群根据式(4)(5)进行重心反向操作得到新种群 $X_{rev}$ ，分别计算种群 $X$ 和 $X_{rev}$ 的适应度，选取适应度较好的个体组成精英种群 $X_{eli}$ ；

Step3:按比例划分发现者、跟随者、警戒者，并分别根据公式(7)、(2)、(3)更新个体位置；

Step4:根据概率轮盘选择混沌映射并对当前种群进行扰动，利用式(10)判断是否扰动成功；若扰动成功，记录成功次数；否则使用式(14)对扰动失败的个体进行高斯变异；

Step5:判断迭代次数是否为 50 的倍数；若判定为真，根据式(12)计算扰动成功率并更新概率轮盘，否则进入下一步；

Step6:判断算法是否满足终止条件，若满足，输出最优解；否则返回 Step3。

## 3 实验仿真与结果分析

### 3.1 测试函数及对比算法

为了验证本文所提 MMCSSA 的有效性和优越性，采用国际上通用的 12 个经典函数 (f1-f12) 进行仿真实验，详细描述如表 1 所示。表中的函数均来自 IEEE CEC(IEEE Congress on Evolutionary Computation)竞赛集<sup>[21,22]</sup>，f1-f4 表示只有一个全局最优解的单峰函数，f5-f12 是指具有多个局部最优解的多峰函数。本文选取 SSA、GWO (Grey Wolf Optimizer,GWO)、文献[11]中的 ISSA (Improved sparrow search algorithm with multi-strategy integration and its application,ISSA) 以及文献[14]中的 CSSA(Chaos sparrow search optimization algorithm,CSSA)作为对比算法进行实验，其中 CSSA 用于对比验证 MMCSSA 逃逸局部极值的能力，ISSA 用于对比验证 MMCSSA 的全局搜索的能力。

Table 1 Benchmark function

表 1 基准测试函数

The serial number	Function name	Domain	Optimal value
f1	Bent Cigar	[-10,10]	0
f2	Sum of Different Power	[-100,100]	0
f3	Zakharov	[-5,10]	0
f4	Rosenbrock's	[-10,10]	0
f5	Rastrigin's	[-5.12,5.12]	0
f6	Levy	[-10,10]	0
f7	Ackley's	[-32,32]	0
f8	Griewank's	[-600,600]	0
f9	Katsuura	[-32,32]	0
f10	Schaffer's F7	[-10,10]	0
f11	Schwefel 2.26	[-500,500]	-418.9d
f12	HappyCat	[-600,600]	0

### 3.2 仿真实验参数设置

为保证实验公平，所有算法均采用相同的参数，种群数目 $N$ 设置为 30，最大迭代次数 $M$ 设为 600，发现者在种群中的数目比例为 0.2，警戒者为 0.1。在  $M$

MCSSA 中，经作者多次实验，多混沌映射策略学习 真实验均在 Windows10 操作系统，MATLAB R2019B 周期按经验设置为 50 代时有最佳的扰动效果。所有仿 仿真平台下进行，每种算法独立运行 30 次。

Table 2 Function test results  
表 2 函数测试结果

Modal	Function	Algorithm	Dim=30		Dim=50		Dim=100	
			Average value	Standard deviation	Average value	Standard deviation	Average value	Standard deviation
Unimodal	f1	GWO	1.37E+10	1.37E+10	1.33E+10	1.33E+10	4.43E+10	1.45E+10
		SSA	1.37E-26	1.37E-26	6.22E-31	6.22E-31	1.64E-27	4.91E-27
		CSSA	4.09E-233	0.00E+00	1.51E-230	0.00E+00	2.54E-229	0.00E+00
		ISSA	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
		MMCSSA	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>
	f2	GWO	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
		SSA	8.53E-259	0.00E+00	1.48E-259	0.00E+00	3.40E-262	0.00E+00
		CSSA	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
		ISSA	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
		MMCSSA	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>
	f3	GWO	4.41E+02	3.56E+02	5.51E+02	3.99E+02	6.52E+01	3.30E+01
		SSA	3.95E-18	2.71E-30	1.30E-57	2.61E-57	6.74E-53	1.35E-52
		CSSA	6.94E-35	0.00E+00	3.37E-76	0.00E+00	8.34E-93	0.00E+00
		ISSA	1.50E-285	0.00E+00	5.40E-319	0.00E+00	0.00E+00	0.00E+00
		MMCSSA	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>
	f4	GWO	1.04E+08	3.96E+07	4.74E+08	6.61E+07	7.81E+07	3.25E+07
		SSA	3.07E-05	6.02E-05	5.23E-05	1.02E-04	2.80E-03	5.29E-03
		CSSA	3.38E-06	3.32E-06	5.66E-05	4.82E-05	1.95E-04	2.55E-04
		ISSA	6.41E-06	4.21E-06	3.55E-05	3.75E-05	1.34E-04	1.85E-04
		MMCSSA	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>
f5	GWO	1.74E+00	2.63E+00	4.71E+00	9.43E+00	9.80E+00	1.96E+00	
	SSA	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	
	CSSA	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	
	ISSA	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	
	MMCSSA	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	
f6	GWO	9.71E-01	2.43E-01	2.39E+00	2.38E-01	5.02E+00	3.25E-01	
	SSA	4.85E-08	5.41E-08	1.74E-07	2.28E-07	9.22E-07	1.69E-06	
	CSSA	1.32E-07	1.17E-07	1.28E-07	2.45E-07	8.84E-08	9.29E-08	
	ISSA	1.87E-08	3.82E-08	3.15E-08	4.67E-08	1.57E-07	2.49E-07	
	MMCSSA	<b>1.29E-12</b>	<b>1.29E-12</b>	<b>1.51E-12</b>	<b>2.76E-12</b>	<b>4.30E-12</b>	<b>6.88E-12</b>	
f7	GWO	5.86E-15	1.74E-15	3.14E-15	7.34E-15	4.35E-15	0.00E+00	
	SSA	8.88E-16	0.00E+00	8.88E-16	0.00E+00	8.88E-16	0.00E+00	
	CSSA	8.88E-16	0.00E+00	-8.88E-16	0.00E+00	-8.88E-16	0.00E+00	
	ISSA	8.88E-16	0.00E+00	-8.88E-16	0.00E+00	-8.88E-16	0.00E+00	
	MMCSSA	<b>8.88E-16</b>	<b>0.00E+00</b>	<b>8.88E-16</b>	<b>0.00E+00</b>	<b>-8.88E-16</b>	<b>0.00E+00</b>	
Multi-modal	f8	GWO	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
		SSA	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
		CSSA	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
		ISSA	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
		MMCSSA	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>
f9	GWO	3.78E-11	2.93E-12	2.75E-11	1.02E-12	1.88E-11	5.28E-13	
	SSA	2.87E-12	4.81E-12	2.17E-13	4.33E-13	2.91E-13	5.82E-13	
	CSSA	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	
	ISSA	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	
	MMCSSA	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	
f10	GWO	5.03E-01	6.15E-01	8.64E-01	3.23E-01	5.70E-01	2.22E-01	
	SSA	3.38E-14	1.01E-13	4.33E-14	8.66E-14	6.13E-14	1.23E-14	
	CSSA	2.45E-16	7.31E-16	2.56E-16	1.34E-16	1.15E-16	5.27E-16	
	ISSA	9.19E-17	1.32E-16	5.22E-17	3.44E-17	1.34E-17	1.09E-17	
	MMCSSA	<b>2.17E-18</b>	<b>5.07E-18</b>	<b>1.11E-18</b>	<b>2.36E-18</b>	<b>8.04E-18</b>	<b>1.65E-18</b>	
f11	GWO	-8.85E+03	<b>1.70E+02</b>	9.12E+03	<b>2.63E+02</b>	-1.16E+04	3.02E+03	
	SSA	-8.87E+03	5.01E+02	1.41E+04	5.55E+03	-9.85E+03	5.58E+03	
	CSSA	-7.98E+03	6.19E+02	1.16E+04	3.94E+02	-9.18E+03	<b>8.57E+02</b>	
	ISSA	-9.35E+03	2.20E+03	1.33E+04	3.11E+03	-1.00E+04	6.21E+03	
	MMCSSA	<b>-1.26E+04</b>	1.46E+03	<b>1.26E+04</b>	3.17E+03	<b>-1.26E+04</b>	1.55E+03	
f12	GWO	4.16E-01	8.01E-02	7.79E-01	6.49E-01	4.12E-01	8.24E-01	
	SSA	1.17E-01	6.74E-02	7.93E-01	1.59E-01	2.10E-01	4.19E-01	
	CSSA	2.91E-02	3.98E-02	5.05E-02	6.12E-02	7.19E-02	8.27E-02	
	ISSA	2.56E-02	1.46E-02	3.64E-02	3.91E-02	1.83E-02	2.93E-02	
	MMCSSA	<b>1.38E-02</b>	<b>8.44E-03</b>	<b>3.11E-03</b>	<b>2.21E-03</b>	<b>7.54E-03</b>	<b>1.29E-02</b>	

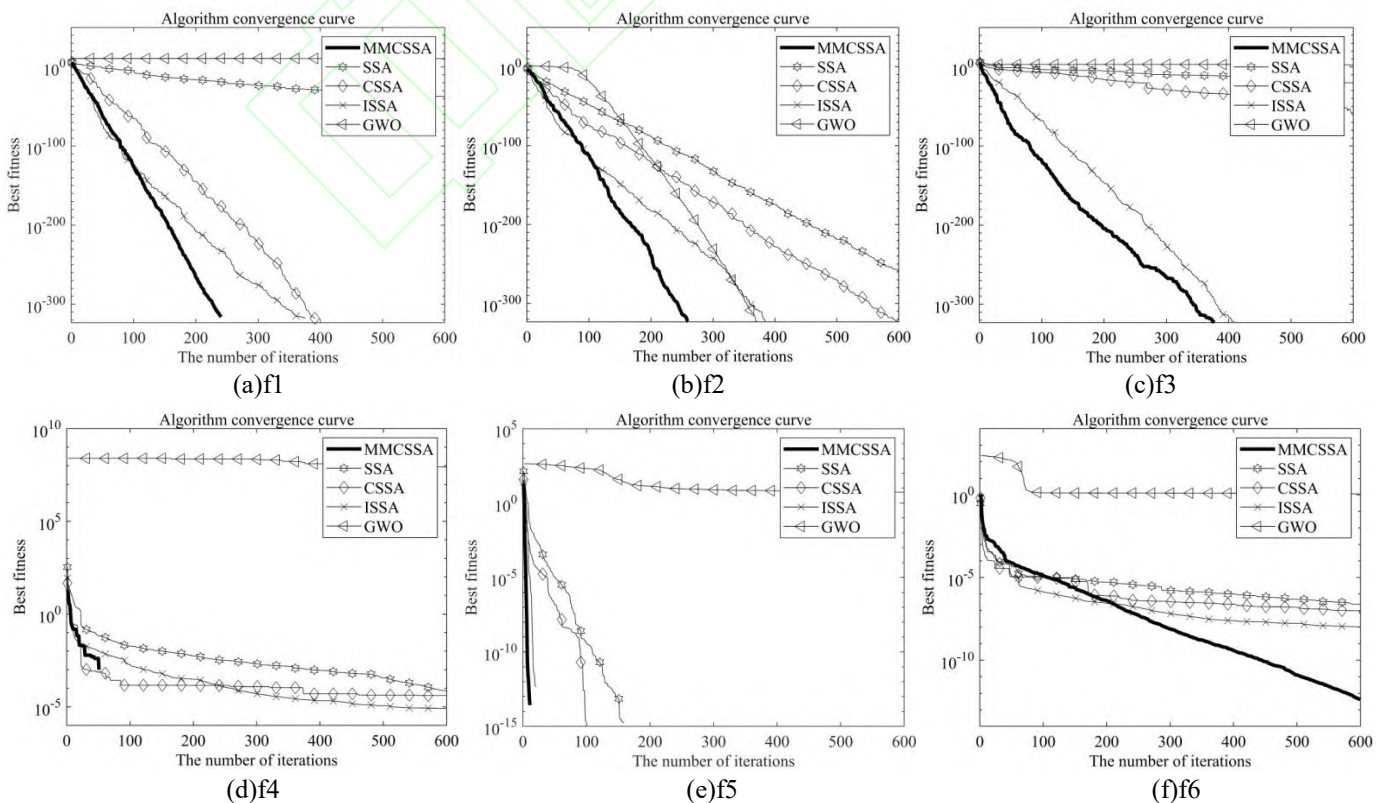
### 3.3 收敛精度与收敛速度分析

表 2 展现了不同算法在 12 个测试函数中不同维度下的实验结果,旨在从单峰/多峰、低维/高维多个角度来分析所提算法的收敛精度。其中,每个函数的最佳值加粗显示。单峰函数 f1、f2、f3、f4 一般用于测试算法的开发能力,由表 2 可以看出,MMCSSA 在单峰函数上的表现优于 CSSA,这是因为单混沌的扰动模式不能帮助 CSSA 在迭代后期跳出局部最优解,而多混沌映射及时选取了合适的扰动模式加强了算法逃逸局部极值的能力;MMCSSA 在 f3、f4 达到了理论最优解且领先其余算法数十个量级,这是因为 MMCSA 的四个改进策略不是单纯的叠加,而是有机互补的,共同提升了算法的收敛精度。另外,MMCSSA 在 f1-f4 的标准差都为 0,这说明 MMCSSA 求解单峰函数时稳定性较好,其背后的原因是将固定趋于原点的移动方式转变为受全局最优位置与个体自身位置共同牵引的移动方式,提升了算法的搜索能力,进而保证了算法的优化稳定性。多峰函数 f5-f12 常用于测试算法的勘探能力,MMCSSA 在 f6、f10、f11、f12 的

收敛精度均优于其他算法,在 f5、f7、f8、f9 上持平于其他改进的 SSA,说明 MMCSSA 在复杂函数上的收敛精度优于其他算法。综上所述,MMCSSA 与其他改进的 SSA 相比,在单峰和多峰测试函数上表现出了更加高的收敛精度。

为了验证 MMCSSA 收敛速度的优越性,图 7 描绘了测试函数的迭代收敛曲线。可以看出,MMCSSA 在收敛速度上同样优于其他算法,具体原因如下:

(1)图 7(d)、(e)、(h)、(i)、(j)中 MMCSSA 的初始适应度值要优于其他算法,说明算法初始化时重心反向学习策略产生的精英种群赋予了算法足够的解空间有效信息。(2)黄金正弦领导策略和多混沌映射策略中分别加入了缩放因子和混沌搜索半径。在迭代后期,缩放因子使种群快速向全局最优解靠拢,搜索半径使混沌扰动效果逐渐减小,两者共同作用进一步提升了算法的收敛速度。(3)多混沌映射策略与高斯变异策略使算法的扰动成功率大大提升,显著地增强了算法的搜索能力,进而促进了算法的快速收敛。





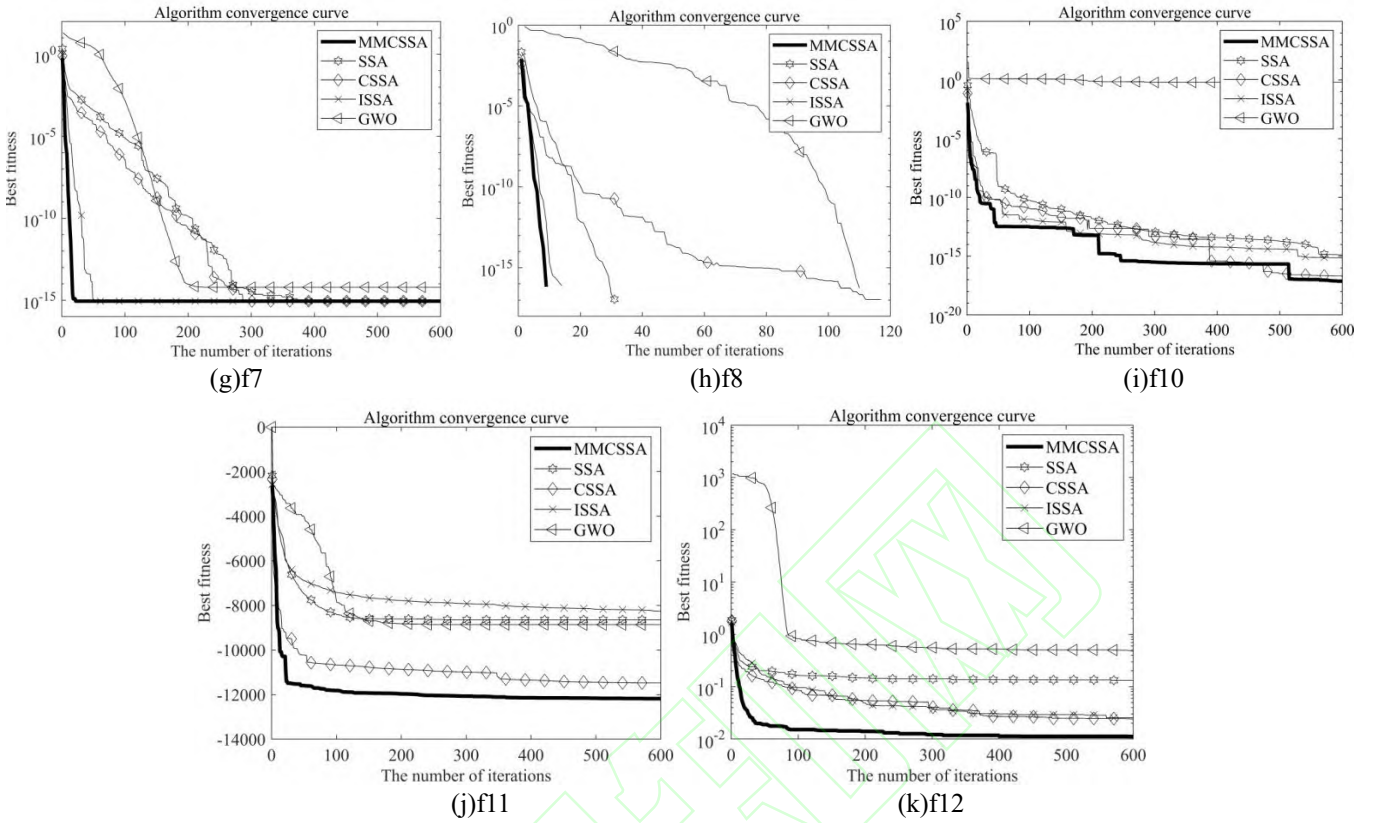


Fig.7 Algorithm convergence curve comparison diagram

图 7 算法收敛曲线对比图

### 3.4 Friedman 检验与 Nemenyi 后续检验

为了更准确地评估每个算法的性能，我们采用 Friedman 检验与 Nemenyi 后续检验<sup>[23]</sup>作为模型的性能评价工具来验证 MMCSA 算法的优越性。

Friedman 检验属于统计假设检验，其基本步骤首先是建立一个假设检验问题：

$H_0$ :  $k$  个算法之间无显著差异

$H_1$ :  $k$  个算法之间存在显著差异。然后在同一个测试函数上按照测试结果对算法性能进行排序，根据性能由好到差对算法依次赋序值。若算法性能相同则将序值平分。结果见表 3。

Table.3 Sequence value table  
表 3 序值表

The serial number	GWO	SSA	CSSA	ISSA	MMCSA
f1	5.0	4.0	3.0	2.0	1.0
f2	2.0	5.0	3.5	3.5	1.0
f3	5.0	4.0	3.0	2.0	1.0
f4	5.0	4.0	2.5	2.5	1.0
f5	5.0	4.0	1.5	3.0	1.5
f6	5.0	2.5	4.0	2.5	1.0
f7	5.0	2.5	2.5	2.5	2.5
f8	5.0	4.0	2.5	2.5	1.0
f9	5.0	4.0	2.0	2.0	2.0
f10	5.0	4.0	3.0	2.0	1.0
f11	3.0	4.0	2.0	5.0	1.0
f12	5.0	4.0	3.0	2.0	1.0
Average ranking	4.6	3.8	2.7	2.6	1.3
Ranking	5	4	3	2	1

平均序值  $r_i$  一般符合自由度为  $k - 1$  的  $\chi^2$  分布，为了获得更准确的检验结果，现使用式(15)进行假设检验， $T_F$  是服从自由度为  $k - 1$  和  $(k - 1)(N - 1)$  的 F 分布（ $N$  为数据集即测试函数数量）。将  $T_F$  的值与 F 检验临界值表中对应的数值进行对比， $T_F$  的值小于对应的数值，则表明“ $k$  个算法之间无显著差异”这一假设被拒绝，即“ $k$  个算法之间存在显著差异”。根据式(15)和(16)计算出  $T_F$  并查表比较，得出“五个算法之间存在显著差异”这一结论。

$$T_F = \frac{(N-1)T_{x^2}}{N(k-1)-T_{x^2}} \quad (15)$$

$$T_{x^2} = \frac{12N}{k(k+1)} \sum_{i=1}^k (r_i^2 - \frac{k(k+1)^2}{4}) \quad (16)$$

为了进一步评价算法间的性能差异，接着使用 Nemenyi 后续检验。根据式(17)计算临界值域  $CD$ ，其中  $\alpha$  表示显著性水平，一般取值为 0.05 和 0.1，本文中取 0.05，即置信度是 95%， $q_\alpha$  为 Nemenyi 后续检验中常用的参数。

$$CD = q_\alpha \sqrt{\frac{k(k+1)}{6N}} \quad (17)$$

通过计算，临界值域  $CD$  为 1.76。若两个算法的平均序值的差值大于  $CD$ ，则以相应的置信度拒绝“两个算法性能相同”这一假设。Friedman 检验图如图 8 所示：

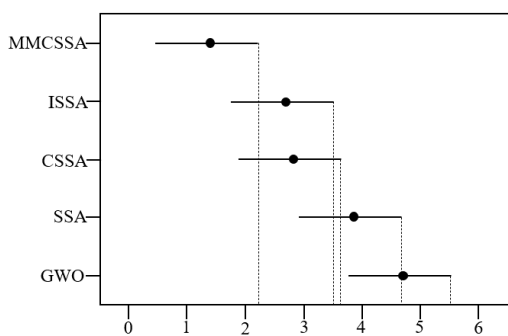


Fig.8 Friedman test diagram  
图 8 Friedman 检验图

图 8 中横轴表示序值, 每个黑点代表其对应算法的平均序值, 线段的长度为临界值域  $CD$ 。纵轴上一个刻度点代表一个算法。若线段末端垂下的虚线与其他线段没有交点, 则表示该线段对应的算法的性能显著优于其余线段对应的算法的性能。由图 8 可知, MMCSSA 的性能显著优于 SSA、GWO, 同时也优于 CSSA 和 ISSA。

## 4 MMCSSA 理论分析

### 4.1 时间复杂度分析

设种群规模为  $N$ , 目标问题的维度为  $D$ , 最大迭代次数为  $T$ 。一般情况下,  $D > \log_2 N$ 。

SSA 的时间复杂度主要由种群初始化、个体位置更新两个部分构成。初始化阶段, 计算  $N$  个个体适应度值的时间复杂度为  $O(N * D)$ , 则初始化阶段的时间复杂度为  $O(N * D)$ ; 迭代中, 种群个体位置更新的时间复杂度为  $O(T(N * D))$ 。综上, SSA 的总时间复杂度为  $O(N * D) + O(T(N * D))$ , 略去低阶项, SSA 的时间复杂度为  $O(T(N * D))$ 。

MMCSSA 的时间复杂度主要由种群初始化、个体位置更新和群体扰动三个部分构成。初始化阶段, 计算  $N$  个个体适应度值的时间复杂度为  $O(N * D)$ , 按 2.1 节对初始种群进行重心反向学习策略的时间复杂度为  $O(N)$ , 选出精英种群的时间复杂度为  $O(N * \log_2 N)$ , 则初始化阶段时间复杂度为  $O(N * D) + O(N) + O(N * \log_2 N)$ , 略去低阶项, 即  $O(N * D)$ ; 迭代中, 种群个体位置更新的时间复杂度为  $O(T(N * D))$ , 按 2.3 节进行多混沌扰动的时间复杂度为  $O(T(N * D))$ , 多混沌扰动结果判别的时间复杂度为  $O(1)$ , 按 2.4 节进行高斯变异时间复杂度为  $O(T(N * D))$ , 则迭代中的时间复杂度为  $O(T(N * D))$ 。综上, MMCSSA 的总时间复杂度为  $O(N * D) + O(T(N * D))$ , 略去低阶项, MMCSSA 的时间复杂度为  $O(T(N * D))$ 。

可以看出 MMCSSA 与 SSA 的时间复杂度相同, 且 MMCSSA 有更好的优化性能。

### 4.2 MMCSSA 全局收敛性分析

MMCSSA 为随机优化算法, 根据算法全局收敛性定理<sup>[24]</sup>, 随机优化算法具有全局收敛性需要满足以下两个条件。

**条件 1**  $f(D(z, \xi)) \leq f(z)$ , 若  $\xi \in S$ , 则  $f(D(z, \xi)) \leq f(z)$ 。

其中  $f$  为最小化问题的目标函数;  $D$  表示随机算法;  $z$  为能够使目标函数的值最小化的解;  $\xi$  为算法  $D$  迭代搜索中的解;  $S$  代表可行解空间。若满足条件 1, 表明目标函数的值有可接受的下确界。

**条件 2** 对于  $S$  中的任意 Borel 子集  $A$ , 若其概率测度  $V[A] > 0$ , 则有

$$\prod_{k=0}^{\infty} (1 - \mu_k(A)) = 0 \quad (18)$$

其中  $\mu_k(A) = P(x^k \in A | x^0, x^1, \dots, x^{k-1})$  为算法  $D$  第  $k$  次迭代的解在集合  $A$  中的概率测度。若满足条件 2, 表明算法经过无穷次迭代后, 不可能错过解空间  $S$  中的任意 Borel 子集  $A$ , 即满足条件的算法无穷次迭代后搜索不到近似全局最优点的概率为 0。

**引理** (随机优化算法全局收敛定理) 若函数  $f$  可测, 搜索空间  $S$  是  $R^n$  上的可测子集, 随机优化算法  $D$  满足条件 1 和条件 2, 则有

$$\lim_{k \rightarrow \infty} P[x^k \in R] = 1 \quad (19)$$

其中  $P[X^k \in R]$  是算法第  $k$  代的结果落在  $R$  里的概率,  $R$  是全局最优解集合, 即算法无穷次迭代后必定搜索到全局最优解。

**定理 1** MMCSSA 满足条件 1

**证明** 根据 MMCSSA 中的描述将  $D$  定义为:

$$D(G_t, X_{i,t}) = \begin{cases} G_t, f(h(X_{i,t})) \geq f(G_t) \\ h(X_{i,t}), f(h(X_{i,t})) < f(G_t) \end{cases} \quad (20)$$

与

$$D(G_t, X_{i,t}) = \begin{cases} G_t, f(y(X_{i,t})) \geq f(G_t) \\ y(X_{i,t}), f(y(X_{i,t})) < f(G_t) \end{cases} \quad (21)$$

其中  $G_t$  为第  $t$  代时全局最优解的位置, 函数  $h$  对动态调整的黄金正弦的位置移动操作,  $h(X_{i,t})$  表示麻雀个体  $i$  按式(7)位置更新后的位置; 函数  $y$  对应混沌扰动和高斯变异操作,  $y(X_{i,t})$  表示麻雀  $i$  进行混沌扰动和高斯变异后的位置。以上操作都使用贪心策略进行位置更新, 按照上述定义, 可知  $G_t$  的适应度值单调不增, 即条件 1 成立。

**定理 2** MMCSSA 满足条件 2

**证明** 对于 MMCSSA 中的自适应黄金正弦领导策略, 有

$$h(X_{i,t}) = w \cdot X_{i,t} \cdot |\sin(r_1)| + r_2 \cdot \sin(r_1) \cdot |w \cdot x_1 \cdot X_{g,t} - x_2 \cdot X_{i,t}| \quad (22)$$

其中  $h(X_{i,t})$  为领导者在  $t$  代时的样本空间。  $h(X_{i,t})$  为  $R^N$  中不同于  $X_{i,t}$  的样本空间, 可知经过以上更新策略, 领导者到  $h(X_{i,t})$  的概率大于 0, 随着迭代次数的

增加，每个  $h(\mathbf{X}_{i,t})$  的闭包  $V[h(\mathbf{X}_{i,t})]$  与其并集  $\cup_{i=1}^N h(\mathbf{X}_{i,t})$  的闭包  $V[\cup_{i=1}^N h(\mathbf{X}_{i,t})]$  都在逐渐变大，故存在一个整数  $t_0$ ，当  $t > t_0$  时

$$S \subseteq \cup_{i=1}^N h(\mathbf{X}_{i,t}) \quad (23)$$

对于 MMCSSA 中麻雀个体经过混沌扰动和高斯变异后，对较差个体进行替代的更新机制，支撑集的并集为  $\delta$ ，随着迭代次数的增加， $V[\delta]$  也在逐渐变大。因此，对于 MMCSSA，存在正整数  $t_1$ ，当  $t > t_1$  时

$$\delta \cup (\cup_{i=1}^N h(\mathbf{X}_{i,t})) \supseteq S \quad (24)$$

令  $S$  的 Borel 子集  $D = \delta \cup (\cup_{i=1}^N h(\mathbf{X}_{i,t}))$ ，则有  $V[D] > 0, \mu_i(D) = \sum_{i=1}^N \mu_{i,t}(D) = 1$ ，即

$$\prod_{t=0}^{\infty} (1 - \mu_t[D]) = 0 \quad (25)$$

条件 2 成立。

由引理可知，MMCSSA 具有全局收敛性。

### 4.3 改进策略的有效性分析

#### 4.3.1 初始化策略的持续影响分析

为分析重心反向学习策略在算法迭代周期的性能影响，分别使用随机初始种群与本文所提初始化方法产生的精英种群来优化 Bent Cigar 函数。该函数在搜索空间内的全局最小值的空间坐标为  $x^* = (0,0,0)$ 。图 (9)(10)(11) 分别给出了两个种群在迭代初期、中期及后期的空间位置分布。图中黑色圆点表示随机初始种群中的个体（随机个体），红色圆点为随机个体经过重心反向学习策略初始化产生的个体（精英个体），蓝色五角星表示种群当前最优解的位置。

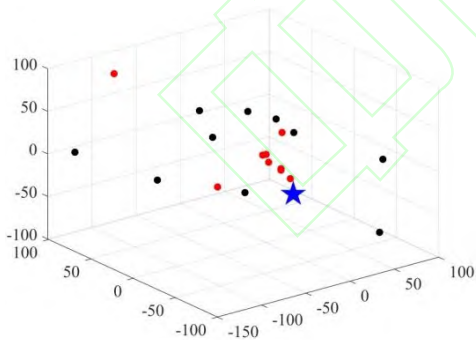


Fig.9 In the early iterations  
图 9 迭代初期

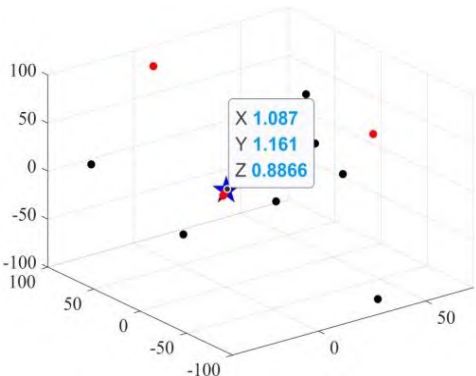


Fig.10 In the mid iterations  
图 10 迭代中期

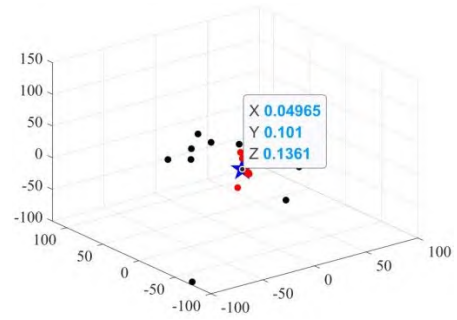


Fig.11 In the late iteration  
图 11 迭代后期

由图 9 可知，迭代初期，精英个体普遍接近全局最优解，说明初始种群经重心反向学习策略与贪心策略双重处理后产生了更加优越的精英个体；由图 10 可以看出，在迭代中期，相比随机个体，精英个体可以更快的接近全局最优解，说明精英个体所代表的种群在迭代周期变迁时仍保持了较好的收敛能力；图 11 表明了种群在迭代后期仍产生了新的最优解，且全局最优位置是由精英个体发现的，一方面，精英个体的位置向量在一定程度上代表优秀的探索方向，可以充分利用种群已有的有效信息；另一方面，精英个体在搜索时会围绕当前最优解不断地挖掘其周边区域的有效信息，两者相互配合，共同推动了算法收敛精度的提高。综上所述，重心反向学习策略持续引导了 MMCSSA 种群的搜索方向，有效提升了算法的收敛精度。

#### 4.3.2 学习机制的多混沌映射理论及有效性分析

混沌序列的概率分布特性和搜索速度是影响混沌扰动的两个主要因素，分别用概率密度函数（probability density function, PDF）和 Lyapunov 指数表示<sup>[25]</sup>。为更清晰地描述不同分布特性对算法产生的扰动效果的影响，除了 2.3.1 节中提到的 7 个混沌映射，表 5 中的实验还加入与 Kent 映射有相同概率分布的 Bernoulli shift 映射作为对照组。表 4 展示了每种混沌映射的函数表达式、使用到的参数以及 Lyapunov 指数。Lyapunov 指数的值都大于 0，表示在区间范围内所有映射均处于混沌状态。

Table 4 Lyapunov exponents  
表 4 Lyapunov 指数

Map	The mapping formula	parameter	Lyapunov exponent
Logistic	$z_{n+1} = \mu z_n(1 - z_n) \quad z \in (0,1) \quad 0 < \mu \leq 4$	$\mu = 4$	0.693
Kent	$z_{n+1} = \begin{cases} z_n / \beta & 0 < z_n \leq \beta \\ (1 - z_n) / (1 - \beta) & \beta < z_n \leq 1 \end{cases}$	$\beta = 0.4$	0.673
Bernoulli shift	$z_{n+1} = \begin{cases} \frac{z_n}{1 - \lambda} & 0 < z_n \leq 1 - \lambda \\ \frac{z_n - (1 - \lambda)}{\lambda} & 1 - \lambda < z_n < 1 \end{cases}$	$\lambda = 0.4$	0.673
Sine	$z_{n+1} = \frac{a}{4} \sin(\pi z_n) \quad z \in (0,1) \quad 0 < a \leq 4$	$a = 4$	0.688
ICMIC	$z_{n+1} = \sin\left(\frac{a}{z_n}\right) \quad z \in [-1,0) \cup (0,1] \quad a \in (0, \infty)$	$a = 2$	1.651
Circle	$z_{n+1} = z_n + \Omega - \frac{k}{2\pi} \sin(2\pi z_n) \bmod 1$	$\Omega = 0.5, k = 2$	0.355
Chebyshev	$z_{n+1} = \cos(k \cos^{-1} z_n) \quad z \in (-1,1)$	$k = 2$	0.6932
Gauss	$z_{n+1} = \begin{cases} 0 & z_n = 0 \\ \frac{\mu}{z_n} \bmod 1 & z_n \neq 0 \end{cases}$	$\mu = 1$	2.377

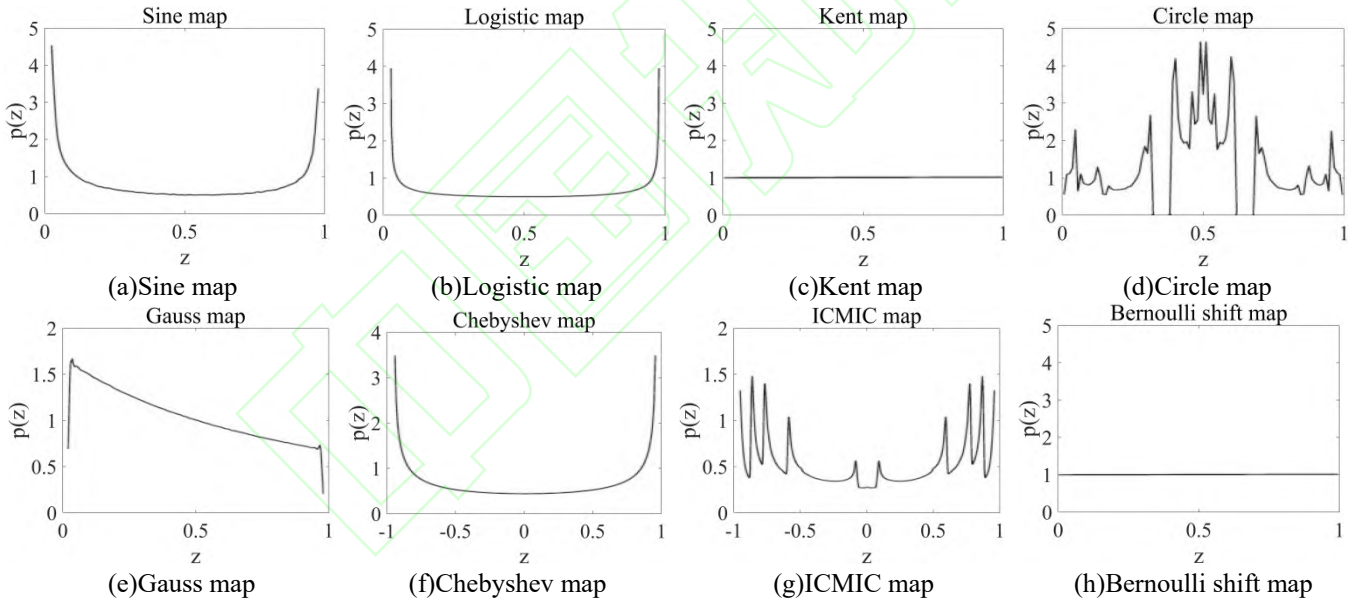


Fig.12 Fixed parameter probability density function diagram

图 12 固定参数概率密度函数图

图 12 展现了不同混沌映射的概率分布情况，图中横轴 $z$ 表示混沌变量的值，纵坐标 $P(z)$ 是 $z$ 对应的概率密度的值。由(b)和(f)可以看出，由于 Logistic 映射与 Chebyshev 映射可以通过线性变换互相转换，导致两种混沌映射的概率分布情况十分相似，仅在变量 $z$ 的区间有所不同。此外，随着参数的不同，混沌映射会同步变化，其对应的概率密度函数也在不断地变化。

为分析概率分布对混沌扰动效果的影响、比较多混沌扰动与单一混沌扰动的性能，选用(1)标准 SSA、(2)单一混沌映射的 SSA(X-SSA)<sup>[25]</sup>、(3)随机选择机制

的多混沌 SSA(Multiple-chaotic sparrow search algorithm with random selection mechanism,RCSSA)<sup>[26]</sup>以及(4)学习选择机制的多混沌 SSA(Multi-chaotic Sparrow search algorithm based on learning mechanism,MMCSSA)四种不同的 SSA 变体对测试函数进行寻优实验。其中，X-SSA 用来分析概率分布与扰动性能的内在联系；X-SSA 与 RCSSA 的比较用于验证多混沌映射策略的有效性；RCSSA 与 MMCSSA 的比较用来验证学习机制的优越性。为了保证实验的公平性，实验结果取一百次实验的平均值，MMCSSA 只使用学习机制的多混沌

映射策略。测试函数选取图 13 中的 5 个基准函数。

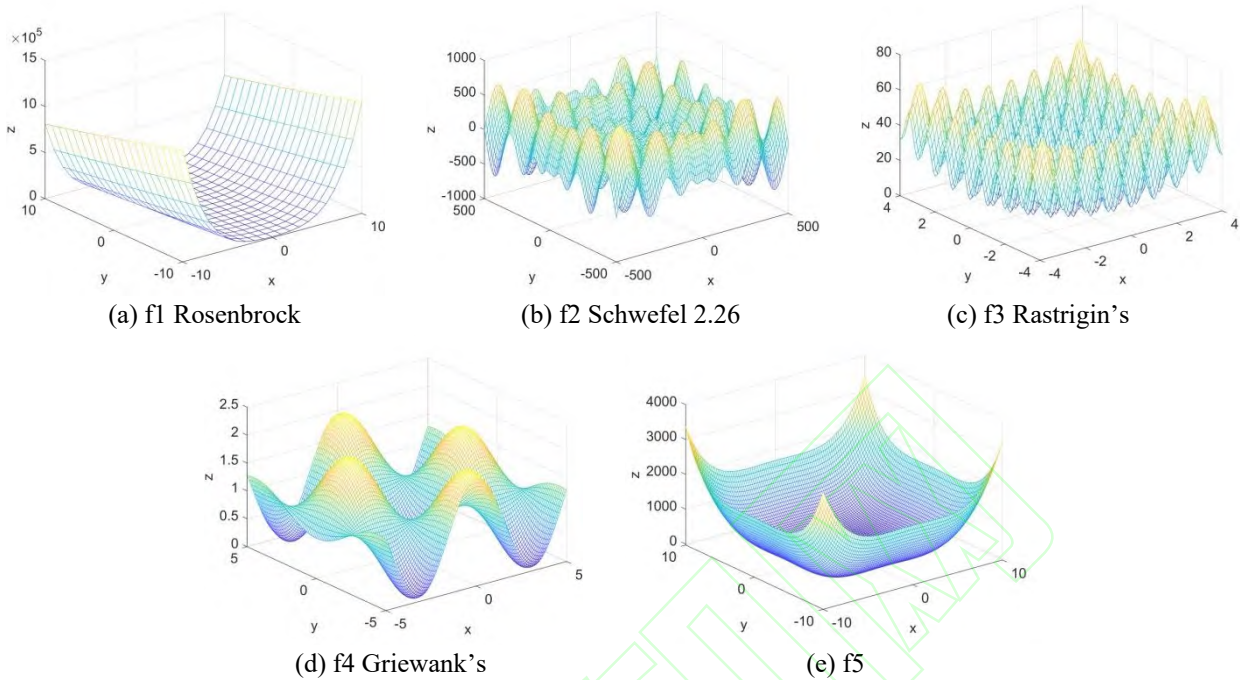


Fig.13 Five test functions

图 13 五个测试函数

函数 f1 Rosenbrock 有一个全局最小值点  $x^* = (1, 1, \dots, 1)$ ，其最优目标函数值为  $f^* = 0$ ；函数 f2 Schwefel 2.26 有多个局部极小值点和一个全局最小值点  $x^* = (420.968, 420.968, \dots, 420.968)$ ， $f^* = -418.98d$ ， $d$  为问题维度；函数 f3 Rastrigin's 有很多局部极小点和一个全局最小值点  $x^* = (0, 0, \dots, 0)$ ， $f^* = 0$ ；函数 f4 Griewank's

s 有多个局部极小值点和一个全局最小值点  $x^* = (0, 0, \dots, 0)$ ， $f^* = 0$ ；函数 f5 有多个局部极小值点和一个全局最小值点  $x^* = (-2.905, -2.905, \dots, -2.905)$ ， $f^* = -78.332$ 。

Table 5 Multi-mode function test results

表 5 基准函数测试结果

Algorithm	f1	f2	f3	f4	f5	Algorithm	f1	f2	f3	f4	f5	Average ranking	Ranking
SSA	1.70E-02	-1.26E+03	2.37E-04	7.42E-08	-7.43E+01	SSA	11	11	11	11	11	11.0	11
LOC-SSA	2.59E-03	<b>-2.09E+03</b>	5.36E-05	2.85E-10	-7.58E+01	LOC-SSA	7	3.5	7	3.5	9	6.0	6
KET-SSA	7.48E-04	-1.89E+03	2.40E-05	3.17E-09	<b>-7.83E+01</b>	KET-SSA	4.5	7	6	6.5	3	5.4	3
BEI-SSA	7.26E-04	-1.84E+03	1.37E-05	7.37E-09	<b>-7.83E+01</b>	BEI-SSA	4.5	8	5	9	3	5.9	5
SIE-SSA	2.30E-03	<b>-2.09E+03</b>	7.26E-05	2.89E-10	-7.58E+01	SIE-SSA	7	3.5	8	3.5	9	6.2	8
ICC-SSA	2.93E-03	<b>-2.09E+03</b>	8.25E-05	1.07E-08	-7.58E+01	ICC-SSA	7	3.5	9	10	9	7.7	10
CIE-SSA	3.75E-04	-1.69E+03	1.77E-06	5.46E-09	-7.73E+01	CIE-SSA	2	10	3	8	6.5	5.9	4
CHY-SSA	6.74E-03	<b>-2.09E+03</b>	1.85E-04	3.46E-09	-7.73E+01	CHY-SSA	9.5	3.5	10	6.5	6.5	7.2	9
GAS-SSA	6.99E-03	-1.78E+03	6.27E-06	1.52E-09	<b>-7.83E+01</b>	GAS-SSA	9.5	9	4	5	3	6.1	7
RCSSA	5.06E-04	<b>-2.09E+03</b>	6.82E-07	7.49E-13	<b>-7.83E+01</b>	RCSSA	3	3.5	2	2	3	2.7	2
MMCSSA	<b>1.42E-06</b>	<b>-2.09E+03</b>	<b>0.00E+00</b>	<b>0.00E+00</b>	<b>-7.83E+01</b>	MMCSSA	1	3.5	1	1	3	1.9	1

表 5 展示了采用不同混沌策略的 SSA 变体对基准函数进行测试的结果。表中 LOC 代表 Logistic；KET 代表 Kent；BEI 代表 Bernoulli；SIE 代表 Sine；ICC 代表 ICMIC；CIE 代表 Circle；CHY 代表 Chebyshev；GAS 代表 Gauss。由表中数据分析可知：

1) 对于单一混沌扰动，KET-SSA、CIE-SSA、BEI-SSA 有较高的性能，LOC-SSA、GAS-SSA、SIE-SSA 为一般性能，CHY-SSA 和 ICC-SSA 为较低性

能。因为 BEI-SSA 和 KET-SSA 用到的混沌映射有相同的概率分布，所以两种算法的性能非常接近，它们之间的细微差异来源于算法的内在随机性。此外，由表 5 可以发现每个混沌映射所适用的函数是不同的，Logistic 映射在函数 f2、f4 上表现良好，对于函数 f5 的表现较差，Gauss 映射在函数 f3、f5 上表现良好，对于函数 f1、f2 的表现较差。

2) 混沌映射的概率密度函数的分布情况与混沌扰动的效果有着直接关系。因为上述五个测试函数的最优解位置是确定的，所以其对应的混沌变量 $Z_i$ 的值也是确定的。对于 f1，全局最优位置对应的混沌变量的值位于(0.4,0.6)区间，而 Circle 映射的概率密度函数的峰值也出现在(0.4,0.6)区间内，这

意味着经过 Circle 映射扰动的个体有较大的概率移动到最优解附近;而 ICMIC 映射的概率分布在(0.4,0.6)区间内的值较小，所以其扰动后的个体出现在最优点附近的概率较低导致混沌扰动的效果较差。

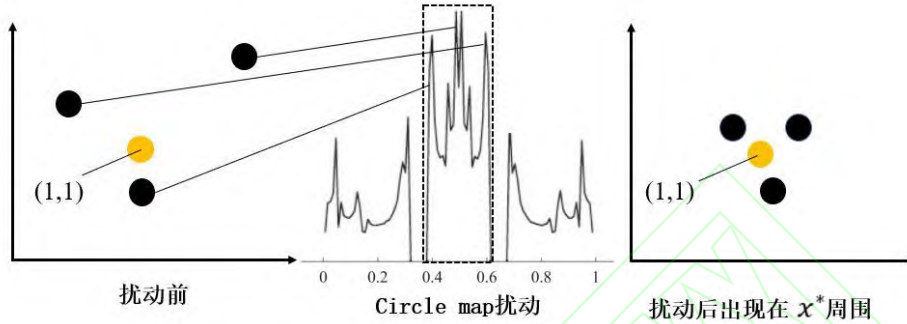


Fig.14 High success rate disturbance  
图 14 高成功率扰动

混沌映射产生序列的值是固定的，如果其值未出现在全局最优解对应的概率分布的区间内，那么算法经混沌扰动后找到全局最优解的概率会大大降低。为克服上述缺陷，需要动态地调整“轨道”。本文使用基于学习机制的概率轮盘动态地选择混沌映射为算法切换不同的“轨道”。下面通过实验来分析学习机制优势

产生的原因。

实验分别使用 RCSSA 和 MMCSSA 求解函数 f1。由于 Bernoulli shift 映射和 Kent 映射具有相同的概率密度函数，其扰动模式重复，所以表 6 和第 3 节中的仿真实验都只使用 Kent 映射。

Table 6 Probability wheel of each stage  
表 6 各阶段概率轮盘

Cycle	MMCSSA/ times	Select probability	RCSSA/ times	Select probability
1	6	The probability of each mapping is 14.28%	5	
2	7	<b>LOC25%KET8%SIE9%GAS24%CHY17%CIE10%ICC7%</b>	0	
3	5	<b>LOC22%KET25%GAS14%CHY15%CIE19%ICC5%</b>	0	
4	15	LOC17%KET14%GAS17%CHY20% <b>CIE32%</b>	3	
5	6	LOC22% <b>KET28%GAS9%CHY15%CIE26%</b>	2	The probability of each mapping is
6	4	LOC17%KET19%GAS11%CHY10% <b>CIE43%</b>	1	14.28%
7	1	KET23%GAS21% <b>CIE56%</b>	2	
8	2	KET19%GAS24% <b>CIE57%</b>	0	
9	1	KET20% <b>GAS69%CIE11%</b>	1	
10	2	KET15% <b>GAS85%</b>	0	

表 6 给出了两种算法在函数 f1 上进行 10 个周期 (Cycle)测试的结果。Times 表示相应算法在每个周期中突破当前全局最优解的次数;Selectprobability 表示每个周期所使用的概率轮盘。由于个体扰动成功次数的作用是设置概率轮盘，并不代表全局最优解得到突破，为了直观地观察每个周期全局最优解被突破的次数，Times 一列中记录了最优位置扰动成功的次数。周期 1 是均匀概率轮盘，每个映射被选到的概率为 14.28%;周期 2 中 Logistic 映射有最高的扰动成功率，而 Logistic 映射并不是最适合求解 f1 的映射，取得这一结果的原因是扰动的内在随机性;周期 3 和周期 4 中 Circle 映射和 Kent 映射对种群扰动的成功率相比周期

2 有显著地提高;在周期 7 与 8 中，Circle 映射逐渐占据了主导地位，这是因为 Circle 映射的概率分布的峰值出现在[0.4,0.6]区间内，实现了高成功率扰动;从周期 9 开始，Gauss 映射扰动的成功率突然增加，而 Circle 映射扰动的成功率快速下降，这种现象产生的原因是 Circle 映射的分布特性已经不适用于算法的收敛阶段，此时需要选择更合适的 Gauss 映射对种群进行持续扰动。以上分析表明，即使是最有效的混沌映射，也不能在算法求解的整体扰动过程中都发挥最大的作用，而本文提出的多混沌映射策略解决了这一缺陷。此外，由 MMCSSA 和 RCSSA 在各阶段的全局最优解被扰动成功的次数可知，本文所提的学习机制的多

混沌映射策略能够有效提升算法的性能。

### 4.3.3 策略之间的互补性分析

本文从三个改进方向提出了下列改进策略：(1)提升初始种群质量的重心反向学习策略、(2)改善发现者位置更新方式的黄金正弦领导策略、(3)增强算法逃逸局部能力的多混沌映射策略和高斯变异策略。提高初始种群的质量是 SSA 提升其收敛精度的重要举措，相比随机种群，重心反向学习策略产生的精英种群因具有更高的质量使其更易发现全局最优解；动态调整的黄金正弦领导策略通过引导个体移动方向改善了 SSA 搜索过程中缺乏方向性的问题，但在遇到特定复杂问题时（如局部最优解与全局最优解在空间上极为接近），种群仍会陷入局部极值；因此，需要对种群进行持续扰动以赋予其多样性来克服上述缺陷，当个体陷入局部最优时，算法首先使用多混沌映射策略在整个解空间中探索优质解，若未发现更优解，说明搜索空间其他区域有更优解的可能性较小，接着使用高斯变异策略在当前区域进行开发以尽最大可能的跳出局部最优解。

综上所述，在 MMCSSA 中，初始化策略产生的精英种群起到提升算法收敛速度的作用；黄金正弦策略能够引导种群进行全局探索；多混沌映射策略和高斯变异策略增强了算法逃逸局部最优的能力，上述四个策略协同作用提高了算法的性能。

### 4.4 MMCSSA 的种群多样性分析

4.3.1 节验证了重心反向学习策略给算法带来的持续性影响，本节为了验证所有改进策略对种群多样性的提升，将 MMCSSA 和 SSA 分别对 Bent Cigar 函数的三维全局最小值点是  $x^* = (0, 0, 0)$ 。下面绘制 MMCSSA 迭代过程中初始种群、迭代 10 次、50 次后的麻雀种群位置分布图和 SSA 相同时期的麻雀种群分布图，图中黑色圆点表示种群个体，红色原点代表当前全局最优个体。以上实验进行 10 次，取代表多次实验的一次结果。

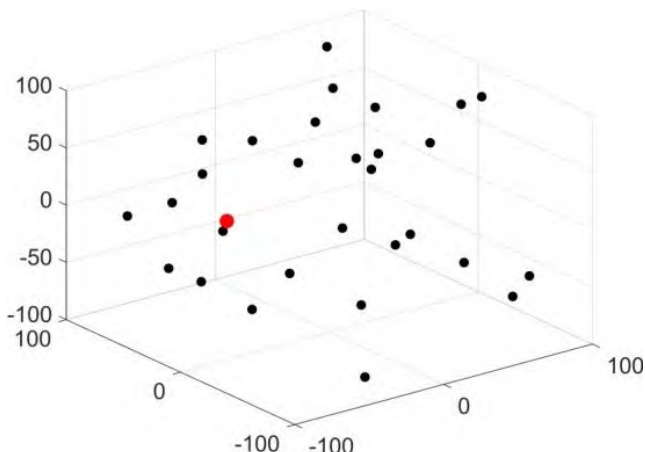


Fig.15 SSA Random initial population

图 15 SSA 随机初始种群

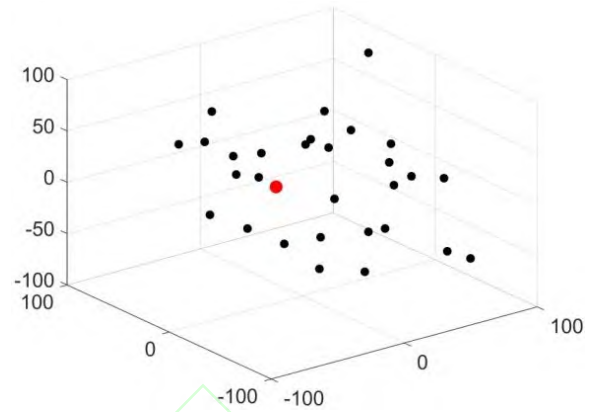


Fig.16 MMCSSA barycentric reverse learning strategy initial population

图 16 MMCSSA 重心反向学习策略初始种群

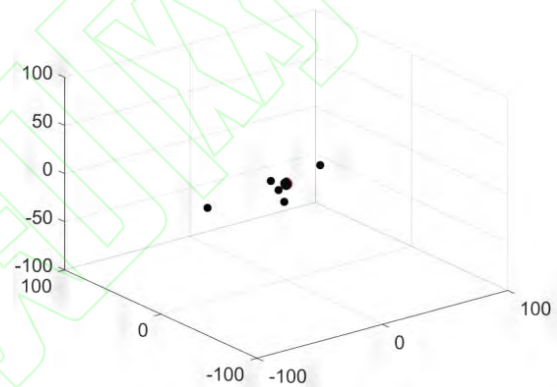


Fig.17 SSA after 10 iterations

图 17 SSA 迭代 10 次后

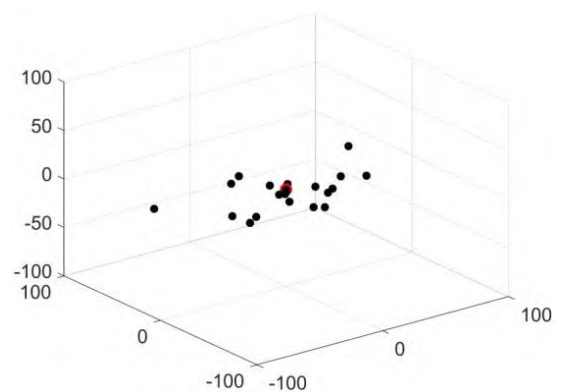


Fig.18 MMCSSA after 10 iterations

图 18 MMCSSA 迭代 10 次后

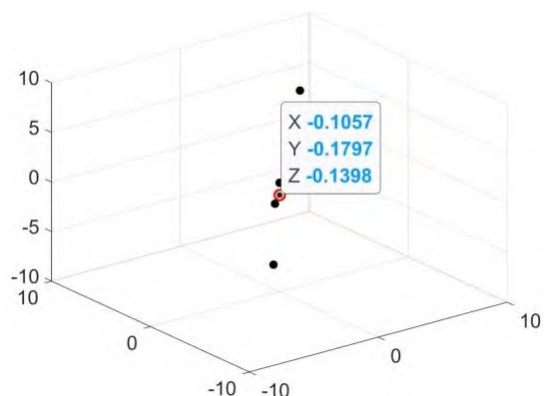


Fig.19 SSA after 50 iterations

图 19 SSA 迭代 50 次后

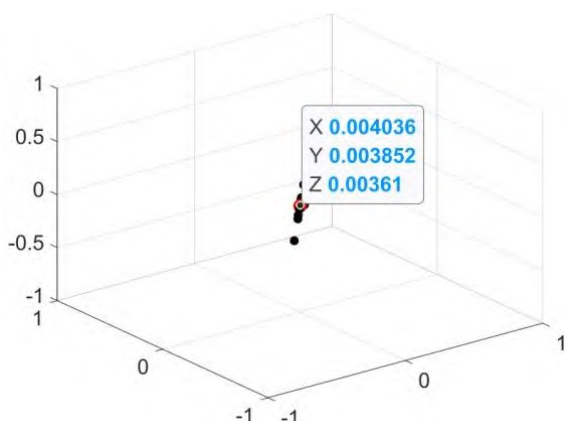


Fig.20 MMCSSA after 50 iterations

图 20 MMCSSA 迭代 50 次后

图 16 中,通过重心反向学习策略初始化得到的精英种群均匀分布在最优解附近;图 17、图 18 对比可知,随着迭代次数的增加,SSA 种群过快的向全局最优位置靠拢,种群多样性急剧降低,MMCSSA 引入黄金正弦领导策略和多混沌映射策略,使得其搜索空间显著大于 SSA;图 19、图 20 对比可知,当迭代到 50 次时,MMCSSA 和 SSA 都集中到全局最小值点(0,0,0)附近,但 MMCSSA 最优解的精度明显高于 SSA,这是由于多混沌扰动帮助算法扩大了搜索空间,找到了更优解。综上,本文提出的 MMCSSA 在种群多样性和搜索空间大小方面均优于 SSA。

## 5 结论

针对麻雀算法易陷入局部极值、后期收敛速度慢等缺点,本文提出了一种融合学习机制的多混沌麻雀搜索算法(MMCSSA),并对 12 个测试函数进行了仿真实验。实验选取了其他学者提出的改进麻雀算法进行比较,并将所得结果进行了统计学分析,最终得到以下结论:

1) MMCSSA 收敛精度高,有较强的全局搜索能力:重心反向学习策略使算法在初始化阶段包含更多解空间的有效信息。迭代过程中,多混沌映射策略和高斯变异策略丰富了种群多样性,有效地帮助算法摆脱局部极值,增强了算法的全局搜索能力。

2) MMCSSA 具有较强的鲁棒性及稳定性:在不同类型、不同维度的函数测试中,相较对比算法,MMCSSA 有着最低的标准差,表现出了较好的优化稳定性。

3) MMCSSA 收敛速度快,优化效率高

黄金正弦领导策略有效调整了算法在收敛速度和搜索能力上的平衡,使算法在更短的时间内达到了更

高的收敛精度。

## References:

- [1] Xue J, Shen B. A novel swarm intelligence optimization approach: sparrow search algorithm[J]. *Systems science & control engineering*, 2020, 8(1): 22-34.
- [2] 王林, 宋蓓, 张友卫, 蔡小龙, 王皓. 考虑父节点的贝叶斯网络故障路径追溯算法[J]. *计算机科学与探索*, 2018, 12(11): 1796-1805.  
WANG L, SONG B, ZHANG Y W, et al. Bayesian network fault path tracing algorithm considering parent node[J]. *Journal of Frontiers of Computer Science and Technology*, 2018, 12(11): 1796-1805.
- [3] 孙怀英, 虞慧群, 范贵生, 陈丽琼. 支持 SDN 的 Hadoop 中的时间最小化任务调度[J]. *计算机科学与探索*, 2018, 12(11): 1767-1776.  
SUN H Y, YU H Q, FAN G S, et al. Support time minimization task scheduling in Hadoop for SDN[J]. *Journal of Frontiers of Computer Science and Technology*, 2018, 12(11): 1767-1776.
- [4] 王振东, 汪嘉宝, 李大海. 一种增强型麻雀搜索算法的无线传感器网络覆盖优化研究[J]. *传感技术学报*, 2021, 34(6): 818-828.  
WANG Z D, WANG J B, LI D H. An enhanced sparrow search algorithm for coverage optimization in wireless sensor Networks[J]. *Chinese Journal of Sensors and Actuators*, 2021, 34(6): 818-828.
- [5] 孟志军, 刘淮玉, 安晓飞, 尹彦鑫, 金诚谦, 张安琪. 基于 SPA-SSA-BP 的小麦秸秆含水率检测模型[J]. *农业机械学报*, 2022, 53(2): 231-238+245.  
MENG Z J, LIU H Y, AN X F, et al. Detection model of wheat straw moisture content based on SPA-SSA-BP[J]. *Transactions of the Chinese Society for Agricultural Machinery*, 2022, 53(2): 231-238+245.
- [6] 毛清华, 张强. 融合柯西变异和反向学习的改进麻雀算法[J]. *计算机科学与探索*, 2021, 15(6): 1155-1164.  
MAO Q H, ZHANG Q. An improved Sparrow algorithm combining Cauchy variation and reverse learning[J]. *Journal of Frontiers of Computer Science and Technology*, 2021, 15(6): 1155.
- [7] 张琳, 汪廷华, 周慧颖. 一种多策略改进的麻雀搜索算法[J/O L]. *计算机工程与应用*: 1-10[2022-06-07]. <http://kns.cnki.net/kcms/detail/11.2127.TP.20220226.1906.010.html>.  
ZHANG L, WANG Y H, ZHOU H Y. A multi-strategy improved sparrow search algorithm [J]. *Computer Engineering and Applications*: 1-10[2022-06-07]. <http://kns.cnki.net/kcms/detail/11.2127.TP.20220226.1906.010.html>.
- [8] 尹德鑫, 张达敏, 蔡朋宸, 秦维娜. 改进的麻雀搜索优化算法及其应用[J/O L]. *计算机工程与科学*: 1-8[2022-04-22]. <http://kns.cnki.net/kcms/detail/43.1258.TP.20210930.1242.002.html>.  
YIN D X, ZHANG D M, CAI P C, et al. Improved sparrow search optimization algorithm and its application[J/O L]. *Computer Engineering & Science*: 1-8[2022-04-22]. <http://kns.cnki.net/kcms/detail/43.1258.TP.20210930.1242.002.html>.
- [9] 张伟康, 刘升, 任春慧. 混合策略改进的麻雀搜索算法[J]. *计算机工程与应用*, 2021, 57(24): 74-82.



- ZHANG W K,LIU S,REN C H. Hybrid strategy improved sparrow search algorithm[J]. Computer Engineering and Applications, 2021,57(24):74-82.
- [10] Rahnamayan S, Jesuthasan J, Bourennani F, et al. Computing opposition by involving entire population[C]. 2014 IEEE Congress on Evolutionary Computation, 2014:1800-1807.
- [11] 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.DOI:10.13195/j.kzyjc.2021.0582.
- FU H,LIU H.Improved sparrow search algorithm based on multi-strategy fusion and its application[J].Control and Decision, 2022,37(1):87-96.DOI:10.13195/j.kzyjc.2021.0582.
- [12] 温泽宇,谢珺,谢刚,续欣莹.基于新型拥挤度距离的多目标麻雀搜索算法[J].计算机工程与应用,2021,57(22):102-109.
- WEN Z Y,XIE JUN,XIE G,et al.Multi-object sparrow search algorithm based on new crowding distance[J].Computer Engineering and Applications,2021,57(22):102-109.
- [13] TANYILDIZI E,DEMIR G.Golden Sine Algorithm: A Novel Math-Inspired Algorithm[J]. Advances in Electrical and Computer,2017,17(2):71-78.
- [14] 吕鑫,慕晓冬,张钧,王震.混沌麻雀搜索优化算法[J].北京航空航天大学学报,2021,47(8):1712-1720.DOI:10.13700/j.bh.1001-5965.2020.0298.
- LV X,MU X D,ZHANG J,et al. Chaotic Sparrow Search Optimization Algorithm[J]. Journal of Beijing University of Aeronautics and Astronautics,2021,47(8):1712-1720. DOI:10.13700/j.bh.1001-5965.2020.0298.
- [15] 李爱莲,全凌翔,崔桂梅,解韶峰.融合正余弦和柯西变异的麻雀搜索算法[J].计算机工程与应用,2022,58(3):91-99.
- LI A L,QUAN L X,CUI G M,et al.Sparrow search algorithm combining sines and cosines with Cauchy variation[J]. Computer Engineering and Applications,2022,58(3):91-99.
- [16] 陈宗淦,詹志辉.面向多峰优化问题的双层协同差分进化算法[J].计算机学报,2021,44(9):1806-1823.
- CHEN Z H,ZHAN Z H. A two-level collaborative differential evolution algorithm for multi-peak optimization problems[J]. Chinese Journal of Computers, 2021,44(9):1806-1823.
- [17] Jia D L, Zheng G X,Khan M K. An effective memetic differential evolution algorithm based on chaotic local search[J]. Information Sciences, 2011, 181(15): 3175-3187.
- [18] Xiang T, Liao X F,Wong K. An improved particle swarm optimization algorithm combined with piecewise linear chaotic map [J]. Applied Mathematics and Computation, 2007,190(2):1637-1645.
- [19] Lijun He, Wenfeng Li, Yu Zhang, Yulian Cao,A discrete multi-objective fireworks algorithm for flowshop scheduling with sequence-dependent setup times[J].Swarm and Evolutionary Computation,2019:51.
- [20] Xianpeng Wang, Lixin Tang,A machine-learning based memetic algorithm for the multi-objective permutation flowshop scheduling problem[J].Computers & Operations Research, 2017, 79: 60-77.
- [21] 何庆,罗仕杭.混合改进策略的黑猩猩优化算法及其机械应用[J/OL].控制与决策,1-11[2022-04-09].DOI:10.13195/j.kzyjc.2021.1108.
- HE Q,LUO S H.Chimp optimization algorithm of hybrid improvement strategy and its mechanical application[J/OL].Control and Decision, 1-11[2022-04-09].DOI:10.13195/j.kzyjc.2021.1108.
- [22] 刘成汉,何庆.融合多策略的黄金正弦黑猩猩优化算法[J/OL].自动化学报,1-14[2022-04-09].DOI:10.16383/j.aas.c210313.
- LIU C H,HE Q.Multi-strategy golden sine chimpanzee optimization algorithm[J/OL].Acta Automatica Sinica, 1-14[2022-04-09].DOI:10.16383/j.aas.c210313.
- [23] 肖辉辉,段艳明,林芳.新搜索策略的花授粉算法[J].电子测量与仪器学报,2019,33(7):11-20.DOI:10.13382/j.jemi.B1801633.
- XIAO H H, DUAN Y M, LIN F. New search strategies for flower pollination algorithms[J].Journal of Electronic Measurement and Instrumentation, 2019,33(7):11-20.DOI:10.13382/j.jemi.B1801633
- [24] Wets J B . Minimization by Random Search Techniques[J]. Mathematics of Operations Research, 1981, 6(1):19-30.
- [25] Yang D, Liu Z, Zhou J. Chaos optimization algorithms based on chaotic maps with different probability distribution and search speed for global optimization[J]. Communications in Nonlinear Science & Numerical, 2014,19(4): 1229-1246.
- [26] S. Gao, Y. Yu, Y. Wang, et al. Chaotic Local Search-Based Differential Evolution Algorithms for Optimization[J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems. 2021, 51(6):3954-3967
- D.X. Yang, G. Li, G.D. Cheng. On the efficiency of chaos optimization algorithms for global optimization[J]. Chaos Solitons Fractals, 2007,34(4), 1366-1375.



LI Guangyang was born in 1997. He is a graduate student of Kunming University of Science and Technology. His research interest is computational intelligence.

李光阳(1997-), 男, 江苏人, 昆明理工大学研究生, 主要研究领域为计算智能。



PAN Jiawen was born in 1994. He is a doctoral candidate of china agriculture university. His research interest are computational intelligence and computer vision.

潘家文(1994-), 男, 山东人, 中国农业大学博士研究生, 主要研究领域为计算智能、计算机视觉。



QIAN Qian was born in 1981. He received the Ph.D degree from Kochi University of Technology in Japan in 2011. Now he is an associate professor at Kunming University of Science and Technology. His research interest includes computational intelligence and visual cognition.

钱谦(1981-), 男, 云南人, 2011 年于日本高知工科大学获得博士学位, 现为昆明理工大学计算机系副教授, 主要研究领域为计算智能与视觉认知。



YIN Jibin was born in 1976. He received his Ph.D degree from Kochi University of Technology in Japan in 2007. Now he is an associate professor in Kunming University of Science and Technology. His research interests include human computer interactions, wearable computing and artificial intelligence.

殷继彬(1976-), 男, 云南人, 2007 年于日本高知工科大学获得博士学位, 现为昆明理工大学计算机系副教授, 主要研究领域为人机交互、可穿戴计算和人工智能。



FU Yunfa was born in 1968. He received the Ph.D degree from Chinese Academy of Sciences in 2012. Now he is a professor at Kunming University of Science and Technology. His research interest includes brain information processing and theory of brain computer interface.

伏云发(1968-), 男, 云南人, 2012 年于中国科学院获得博士学位, 现为昆明理工大学计算机系教授, 主要研究领域为脑信息处理与脑机接口理论。



FENG Yong was born in 1975. He received the Ph.D degree from University of Electronic Science and Technology of China in 2011. Now he is a professor at Kunming University of Science and Technology. His research interest includes block chain and vehicular Ad-hoc network.

冯勇(1975-), 男, 云南人, 2011 年于电子科技大学获得博士学位, 现为昆明理工大学计算机系教授, 主要研究领域为区块链和车载自组织网络。