

基于动态反向学习和黄金正弦策略的麻雀算法研究

杨晓芳¹, 蔡鑫¹, 徐徐¹, 高薪越¹, 孔令琦², 蒋光慧¹

¹盐城工学院信息工程学院, 江苏 盐城

²智洋创新科技股份有限公司, 山东 济南

收稿日期: 2023年10月16日; 录用日期: 2023年12月12日; 发布日期: 2023年12月20日

摘要

针对麻雀算法存在的收敛速度慢, 寻优精度不高, 易陷入局部最优的问题, 提出一种基于动态反向学习和黄金正弦策略改进的麻雀算法。引入动态反向学习策略生成初始种群, 从而扩大搜索空间, 增加多样性。此外, 引入黄金正弦因子来提高种群个体质量, 以改善算法性能, 并采用多项式变异扰动与双面镜反射理论边界优化策略, 以增强跳出局部最优解的能力。为了验证新算法优化性能和普适性, 选用10个标准测试函数与其他优化算法进行比较。实验结果表明, 相对于传统的麻雀算法、粒子群算法以及灰狼算法, 本文提出的改进算法具有更好的收敛精度和更快的收敛速度, 在性能上表现出一定的优越性。

关键词

麻雀算法, 动态反向学习, 黄金正弦因子, 多项式变异扰动, 双面镜反射理论边界优化

Research on Sparrow Search Algorithm Based on Dynamic Opposite Learning and Golden Sine Strategy

Xiaofang Yang¹, Xin Cai¹, Xu Xu¹, Xinyue Gao¹, Lingqi Kong², Guanghui Jiang¹

¹School of Information Engineering, Yancheng Institute of Technology, Yancheng Jiangsu

²Zhiyang Innovation Technology Company Limited, Jinan Shandong

Received: Oct. 16th, 2023; accepted: Dec. 12th, 2023; published: Dec. 20th, 2023

Abstract

In this paper, an improved Sparrow Search Algorithm (DGSSA) based on Dynamic Opposite Learning

文章引用: 杨晓芳, 蔡鑫, 徐徐, 高薪越, 孔令琦, 蒋光慧. 基于动态反向学习和黄金正弦策略的麻雀算法研究[J]. 运筹与模糊学, 2023, 13(6): 6827-6836. DOI: 10.12677/orf.2023.136671

and Golden Sine strategy is presented to address the limitations of the standard Sparrow Search Algorithm (SSA), with the aim of improving convergence speed, optimization precision, and overcoming local optimization issues. The improved method includes the introduction of dynamic opposite learning strategy to generate the initial population, thereby expanding the search space and increasing the diversity. In addition, the Golden-Sine factor is introduced to enhance individual mass within the population, thereby improving the algorithm's performance. Moreover, the polynomial variation and double-faced mirror reflection theory boundary optimization strategy are introduced to improve the capability of escaping local optimal solutions. To verify that the DGSSA optimizes performance and universality, this paper chooses 10 standard test function comparing with other optimization algorithms. The experimental results show that compared with SSA, particle swarm algorithm and gray wolf algorithm, the DGSSA algorithm has better convergence accuracy and faster convergence speed, and has presented in this paper exhibits notable performance advantages.

Keywords

Sparrow Search Algorithm, Dynamic Opposite Learning, Golden Sine Factor, Polynomial Variation, Double-Faced Mirror Reflection Theory Boundary Optimization

Copyright © 2023 by author(s) and Hans Publishers Inc.

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

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



Open Access

1. 引言

在现实生活中，人类面对众多复杂的优化问题。传统解决方法如牛顿法和梯度下降法在处理这些问题时在局部搜索方面表现不佳，难以跳出局部最优解，因而无法满足挑战性的优化任务需求。随着科学研究的不断发展，越来越多的群体智能优化算法应运而生。这些算法基于生物的生命特征或行为规则，建模出一系列数学公式，并根据这些公式在特定的解空间中寻找高质量的解。近年来，由于这些算法具有简便性、对初始值不敏感等特点，它们在路径规划、信号处理等各个领域受到了广泛关注。典型的群体智能优化算法包括麻雀优化算法[1]、灰狼优化算法[2]、粒子群算法[3]、鲸鱼优化算法[4]、人工蜂群算法[5]、哈里斯鹰优化算法[6]、斑马优化算法[7]以及淘金优化算法[8]等。群智能算法的实施为解决复杂的优化问题提供了新的途径，具有重要的理论和实际意义。

在 2020 年，Xue 等提出了一种新的元启发式算法麻雀优化算法(SSA)。该算法模拟麻雀的觅食与反捕食行为方式而来。这一算法不仅具备较少的群体角色，而且其原理简单易懂。与灰狼优化算法(GWO)和粒子群优化算法(PSO)相比，SSA 算法表现出更高的精度、更强的收敛性和更卓越的优化能力[1]。因此，SSA 算法在实际工程中得到了广泛应用。然而，SSA 算法也存在一些群体智能算法的普遍缺点，如容易陷入局部最优解、初始化过程中随机性较大等问题。为了提升和优化算法性能，研究人员提出了各种改进方法。例如，杜云等人提出了混合多项自适应权重的混沌麻雀搜索算法，通过增加混沌映射来提高种群多样性，并引入自适应权重策略，以增强全局搜索能力和搜索范围[9]。张恩辅等人提出了一种菱形分组机制，以增加算法种群的多样性，同时引入融合模拟退火思想的方法，以在一定概率下接受“较差”的结果，从而改善算法容易陷入局部极值点的问题[10]。Ouyang 等人提出了自适应螺旋飞行麻雀优化算法，将鲸鱼优化算法的气泡网捕食策略与麻雀优化算法有机融合，使得种群个体的位置更新更加灵活，从而使算法更好地避免局部最优解的困境[11]。这些改进方法丰富了 SSA 算法的应用领域，并提高

了其在实际问题中的性能。

为了提高麻雀优化算法的性能, 本文将动态反向学习策略和多项式变异扰动相结合, 同时引入黄金正弦因子和双面镜反射理论边界优化策略, 提出了一种基于动态反向学习和黄金正弦策略改进的麻雀算法(DGSSA), 相比较于原麻雀优化算法, DGSSA 在寻优精度和收敛速度上得到了明显的提升。

2. 麻雀优化算法

麻雀优化算法(SSA)受到自然界中麻雀鸟群的觅食行为启发。该算法模拟了麻雀群体的协同寻找食物的策略, 分为探索者和跟随者两类角色。探索者引领群体寻找食物, 而跟随者紧随其后, 通过信息传递和协作找到最佳的食物源。麻雀个体之间的角色可能交换, 但比例保持不变。部分麻雀个体还保持警惕, 以预防潜在的危险。当发现天敌到来时, 整个群体会在探索者的带领下移动, 并寻找新的食物源。

在该模型中规定, 那些适应性更佳的麻雀将被指派为探索者, 在搜索过程中将获得优先获取食物资源。因此探索者位置的更新公式如下:

$$x_{i,j}^{t+1} = \begin{cases} x_{i,j}^t + Q \cdot L & \text{if } R_2 \geq ST \\ x_{i,j}^t \cdot \exp\left(\frac{-i}{R_1 \cdot T}\right) & \text{if } R_2 < ST \end{cases} \quad (1)$$

其中, t 表示当前的迭代次数, j 取值于 $\{1, 2, \dots, D\}$, D 为解的维度。 $x_{i,j}$ 表示第 i 个麻雀在第 j 维度上的取值。 T 表示最大迭代次数, R_1 和 R_2 是两个独立随机数, 取值范围都是 $(0, 1]$, ST 为安全阈值。 Q 是一个服从标准正态分布的随机数, L 是一个元素都为 1 的 $1 \times D$ 矩阵。

跟随者会根据探索者适应度值的优劣确定寻觅的位置, 其位置更新公式如下:

$$x_{i,j}^{t+1} = \begin{cases} Q \cdot \exp\left(\frac{x_{worse}^t - x_{i,j}^t}{i^2}\right) & \text{if } i > \frac{n}{2} \\ x_p^{t+1} + |x_{i,j}^t - x_p^{t+1}| \cdot A^+ \cdot L & \text{if } i \leq \frac{n}{2} \end{cases} \quad (2)$$

其中, x_p 代表当前生产者中适应度最高的个体位置, x_{worse} 表示全局适应度最劣的个体位置, A 是一个包含 1 与 -1 随机分布的 $1 \times D$ 矩阵, A^+ 表示 $A^T (AA^T)^{-1}$ 。

在每次麻雀种群迭代更新的过程中, 部分麻雀会察觉到危险的存在, 它们从种群中随机选出, 并进行位置的更新, 更新公式如下:

$$x_{i,j}^{t+1} = \begin{cases} x_{best}^t + \beta \cdot |x_{i,j}^t - x_{best}^t| & \text{if } f_i > f_g \\ x_{i,j}^t + K \cdot \left(\frac{|x_{i,j}^t - x_{worst}^t|}{(f_i - f_w) + \varepsilon} \right) & \text{if } f_i \leq f_g \end{cases} \quad (3)$$

其中, x_{best} 表示种群中最优的个体位置。 β 是一个标准正态分布的随机数。 K 是一个属于 $[-1, 1]$ 的随机数。 f_i 、 f_g 和 f_w 分别表示为当前个体的适应度值、当前全局最优适应度值以及当前最劣适应度值。 ε 是一个趋近于 0 的常数。

3. 改进麻雀优化算法

3.1. 动态反向学习种群初始化

本文通过采用动态反向学习策略(DOL)来增加种群的初始多样性[12], 其初始化描述如下:

$$oP_{i,j} = lb_j + rand \cdot (ub_j - lb_j) \tag{4}$$

$$oP_{i,j}^{DO} = oP_{i,j} + r1_i \cdot (r2 \cdot (lb_j + ub_j - oP_{i,j}) - oP_{i,j}) \tag{5}$$

其中 oP 是随机获得的正常初始群体, oP^{DO} 是由 DOL 获得的一个群体, 它比正常初始种群解更优质。在任何迭代 t 处, 总体大小被设置为 N (对于任何个体 i , $i = \{1, 2, \dots, N\}$), 解维度被定义为 D (对于任何维度 j , $j = \{1, 2, \dots, D\}$)。 $[lb, ub]$ 是解的边界, $r1_i$ 和 $r2_i$ 是属于 $[0, 1]$ 的随机数。此外, 还需按公式(6)进行检查边界。最后从 $\{oP \cup oP^{DO}\}$ 中选择前 N 个适应度最优的个体作为初始化种群 x 。

$$oP_{i,j}^{DO} = rand(lb_j, ub_j) \quad \text{if } oP_{i,j}^{DO} < lb_j \parallel oP_{i,j}^{DO} > ub_j \tag{6}$$

相比随机初始化方法, 基于动态反向学习生成的初始化种群有着较好的分布。

3.2. 动态反向学习迭代跳跃

通过考虑跳跃率 Jr (Jr 属于 $(0, 1)$ 区间), 利用 DOL 更新种群[12]。在每次迭代中, 生成一个 $[0, 1]$ 区间内的随机数作为选择概率 $r5$, 如果 $r5$ 小于跳跃率 Jr , 则需要按照公式(7)得到动态反向解。

$$oP_{i,j}^{DO} = x_{i,j}(t) + r3_i \cdot (r4_i \cdot (a_j + b_j - x_{i,j}(t)) - x_{i,j}(t)) \tag{7}$$

为了使 DOL 有效, 需要按公式(8)的形式进行检查边界。 $r3$ 和 $r4$ 分别表示一个 $1 \times D$ 的矩阵, 各元素随机分配 $[0, 1]$ 区间内的随机数, 边界 $[a, b]$ 需要按公式(9)进行动态更新, 将新的候选锁定在缩小的搜索空间内。从 $\{x(t) \cup oP^{DO}\}$ 中选择前 N 个适应度最优的个体作为子代种群 $x(t+1)$, 并且迭代次数加 1, 表明进行了一次迭代跳跃。

$$oP_{i,j}^{DO} = rand(a_j, b_j) \quad \text{if } oP_{i,j}^{DO} < a_j \parallel oP_{i,j}^{DO} > b_j \tag{8}$$

$$\begin{cases} a_j = \min(x_{i,j}(t)) \\ b_j = \max(x_{i,j}(t)) \end{cases} \tag{9}$$

由于传统的反向学习策略都存在将搜索空间收敛到该局部最优位置的问题, 如果从当前决策变量到相反决策变量的空间中存在局部最优, 则这些策略都有将搜索空间收敛到该局部最优位置的趋势, 相对于传统反向学习策略, DOL 策略既能提高收敛到全局最优的概率, 又能避免同步陷入局部最优[12]。因此将 DOL 引入麻雀优化算法可加快收敛速度, 提高全局优化能力。

3.3. 黄金正弦因子

黄金正弦算法是由 Tanyildizi 等学者于 2017 年提出的优化算法, 该算法基于正弦函数和单位圆的关系求解优化问题, 同时引入黄金分割数来缩小解空间, 以便快速扫描潜在的优化区域[13]。这不仅提高了收敛速度, 还平衡了全局和局部搜索的能力, 从而提高了优化精度。

在探索者阶段加入黄金正弦因子, 则位置更新公式如下:

$$x_{i,j}^{t+1} = \begin{cases} x_{i,j}^t \cdot |\sin(r1)| - r2 \cdot \sin(r1) \cdot |m \cdot x_{worst}^t - n \cdot x_{i,j}^t| & \text{if } R_2 < ST \\ x_{i,j}^t + Q \cdot L & \text{if } R_2 \geq ST \end{cases} \tag{10}$$

其中, $r1$ ($r1 \in [0, 2\pi]$) 和 $r2$ ($r2 \in [0, \pi]$) 为随机数, m 和 n 是采用黄金分割法后产生的系数, 表达式如下:

$$\begin{cases} m = \alpha \cdot (1 - \tau) + \beta \cdot \tau \\ n = \alpha \cdot \tau + \beta \cdot (1 - \tau) \end{cases} \quad (11)$$

公式(11)中, τ 是黄金分割数, 取值为 $(1 - \sqrt{5})/2$, α 和 β 分别为 $2 - 4t/T$ 和 1 , t 当前迭代次数, T 为最大迭代次数。

3.4. 多项式变异扰动

传统的麻雀算法面临着迭代后期种群多样性减少、搜索能力下降等问题。为此, 本文引入了多项式变异扰动策略[14], 对麻雀位置进行更新, 保持了种群的多样性, 提高了麻雀算法跳出局部最优的能力, 在搜索后期也进一步加快了算法收敛速度, 增强了搜索效率。具体变异扰动算子形式如下:

$$x(t+1) = x(t) + \omega \cdot (ub - lb) \quad (12)$$

其中, ω 、 δ_1 、 δ_2 由公式(13)~(15)表示, u 表示 $[0, 1]$ 上的随机数, η_m 是一个非负实数, lb 和 ub 分别表示决策变量的上边界与下边界, $x(t)$ 表示一个父代个体, $x(t+1)$ 变异后的新个体。

$$\omega = \begin{cases} \left[2u + (1 - 2u)(1 - \delta_1)^{\eta_m+1} \right]^{\frac{1}{\eta_m+1}} - 1 & \mu \leq 0.5 \\ 1 - \left[2(1 - u) + 2(u - 0.5)(1 - \delta_2)^{\eta_m+1} \right]^{\frac{1}{\eta_m+1}} - 1 & \mu > 0.5 \end{cases} \quad (13)$$

$$\delta_1 = (x(t) - lb) / (ub - lb) \quad (14)$$

$$\delta_2 = (ub - x(t)) / (ub - lb) \quad (15)$$

3.5. 双面镜反射理论边界优化

当个体的解超出问题的边界时, 通常会将其值限制在上下边界, 这可能导致解在边界附近聚集, 而在其他区域分布稀疏, 这可能会对算法的性能产生不利影响。

为克服这一问题, 本文采用了双面镜反射边界优化方法[15]。这个方法将决策变量的上下边界类比为镜子, 将 y 视为光束, y 的大小表示光的强度。如图 1 所示, 随着光束的多次反射, 最终在边界内的 y' 位置消失, 所以将 y' 视为 y 在边界内的投影。通过这一方法, 能够有效地解决传统边界处理导致的分布不均问题。具体的优化方法如下所示:

$$y = \begin{cases} ub - \text{mod}(y - ub, ub - lb) & y > ub \\ lb + \text{mod}(lb - y, ub - lb) & y < lb \end{cases} \quad (16)$$

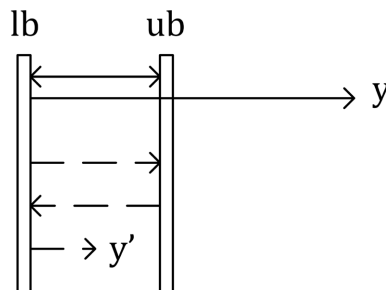


Figure 1. Double-faced mirror reflection schematic
图 1. 双面镜理论示意图

3.6. DGSSA 算法整体流程

DGSSA 首先通过 DOL 策略进行种群初始化。探索者通过黄金正弦因子提高收敛速度，并平衡全局和局部搜索的能力。引入多项式变异扰动，防止跟随者聚集促使麻雀个体跳出局部最优。其次双面镜反射理论边界优化机制有效地解决传统边界处理所导致分布不均的问题。DGSSA 算法的伪代码如下：

算法： DGSSA 伪代码

输入：

N ：麻雀种群规模
 PD ：麻雀探索者数量
 SD ：意识到危险的麻雀数量
 R_2 ：警报值
 T ：最大迭代次数
 定义相关系数并利用动态反向学习策略初始化种群

输出：

X_{best} ：全局最佳个体位置
 f_g ：全局最优适应度值

```

1: while ( $t < T$ )
2: 对适应度值进行排名并找到当前最优值个体和当前最差的个体
3:    $R_2 = rand(1)$ 
4:   for  $i = 1: PD$ 
5:     按照公式(10)更新麻雀位置
6:   end for
7:   for  $i = (PD + 1): N$ 
8:     按照公式(2)更新位置麻雀位置
9:   end for
10:  for  $i = 1: SD$ 
11:   按照公式(3)更新位置麻雀位置
12: end for
13: if 算法陷入停滞
14:   按照公式(12)对麻雀种群进行变异操作
15: end if
16:   if  $rand < Jr$ 
17:     按照公式(7)~(9)进行动态反向学习操作
18:    $t = t + 1$ 
19:   end if
20: 获取子代种群位置
21: 如果子代种群位置的适应度值优于父代种群适应度位置，进行替换
22:    $t = t + 1$ 
23: end while
24: return  $X_{best}, f_g$ 
  
```

4. 实验结果与分析

为检验本文所阐述算法的可行性与有效性，在 Window11 64 位操作系统和 32 GB 内存的计算机上进行了一系列仿真实验。将文中提出的多策略融合的麻雀优化算法(DGSSA)与麻雀算法(SSA)、粒子群优化算法(PSO)、灰狼优化算法(GWO)进行对比实验，并分析它们的平均收敛曲线。选取了 CEC2005 测试集中的十个基准测试函数，特征为单峰 F1~F5，特征为多峰 F6~F10，在表 1 中列出了它们的详细信息。其

中所有算法种群规模为 30，迭代次数为 500，并将每个测试函数独立运行 30 次，最后取平均值和标准差作为结果。表 2 展示了各种算法的函数平均值及标准差。如图 2(a)~(j)展示了十个标准测试函数的优化收敛效果。

Table 1. Test set function

表 1. 测试集函数

函数	函数名称	范围	F_{\min}	函数	函数名称	范围	F_{\min}
F1	<i>Sphere</i>	$[-100,100]$	0	F6	<i>Schwefel2.26</i>	$[-500,500]$	-418.983
F2	<i>Schwefel2.22</i>	$[-10,10]$	0	F7	<i>Ackley</i>	$[-32,32]$	0
F3	<i>Schwefel1.2</i>	$[-100,100]$	0	F8	<i>Shekel's Foxholes</i>	$[-65.53,65.53]$	0.998
F4	<i>Schwefel2.21</i>	$[-100,100]$	0	F9	<i>Kowalik</i>	$[-5,5]$	$3.07e-4$
F5	<i>Quartic</i>	$[-1.28,1.28]$	0	F10	<i>Goldstein-Price</i>	$[-2,2]$	3

由图 2 明显可见，DGSSA 算法的收敛速度与寻优精度显著优于其他几种算法，尤其是在单峰函数 F1~F5 的测试中。在图 2(f)可以看出，在多峰函数 F6 的测试中，虽然 DGSSA 的收敛速度略慢于 SSA 算法，但其寻优精度明显更高。此外，从图 2(g)~(j)的结果可看出，在多峰函数 F7~F10 的测试中，DGSSA 在不到 100 次迭代后就能够收敛到最佳值，其收敛速度明显快于 SSA 等其他算法。总而言之，通过对十种基准测试函数的收敛曲线分析，本文所提出的 DGSSA 相较于 SSA，在解决复杂问题时具有更快的收敛速度和更强的全局搜索能力。

在表 2 中，平均值、标准差反映了 DGSSA 和其他优化算法对十种基准测试函数所求解的质量。标准差反映出算法的稳定性，尤其针对单峰测试函数，DGSSA 均有较明显优势。对于多峰函数的求解 DGSSA 比其他算法要更加稳定。总的来看，从平均值和标准差两个角度看，DGSSA 相对于 SSA 表现出更小的值，这表明在全局寻优能力和稳定性方面，DGSSA 比 SSA 更具优势。实验结果表明，本文改进的麻雀搜索算法在提高寻优精度和加快收敛速度方面表现显著。究其原因，在于动态反向学习和多项式变异扰动策略有助于保持种群的多样性，同时利用黄金正弦因子加快算法收敛速度与收敛精度。

Table 2. Comparison of test results

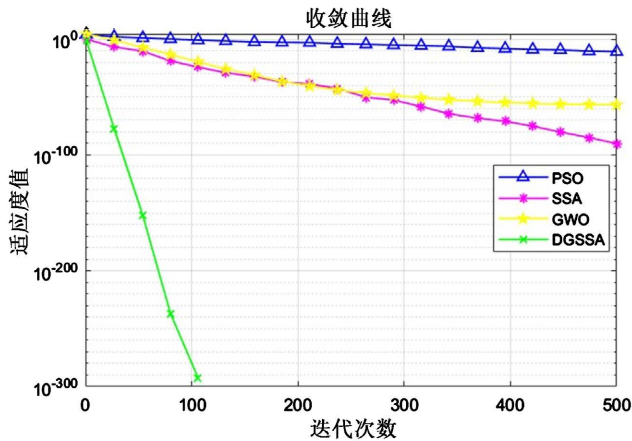
表 2. 测试结果对比

测试函数	GWO		PSO		SSA		DGSSA	
	平均值	标准差	平均值	标准差	平均值	标准差	平均值	标准差
F1	5.45E-62	2.70E-56	5.10E-12	5.16E-10	8.90E-162	1.30E-67	0	0
F2	3.56E-35	1.21E-32	1.44E-07	1.32E-6	3.20E-119	8.04E-36	0	0
F3	8.93E-31	6.75E-25	1.49E-4	4.52E-3	0	3.54E-33	0	0
F4	2.21E-20	5.32E-18	2.69E-4	1.88E-3	7.60E-243	5.35E-32	0	0
F5	6.01E-5	4.21E-4	1.02E-3	1.59E-3	4.21E-4	1.11E-3	3.54E-6	1.55E-4
F6	-3.42E+3	3.43E+2	-3.52E+3	1.82E+2	-3.95E+03	2.89E+2	-4.19E+3	2.91E+2
F7	4.00E-15	1.94E-15	8.93E-7	5.85E-6	4.44E-16	0	4.44E-16	0
F8	9.98E-1	3.72	9.98E-1	2.48	9.98E-1	5.24	9.98E-1	3.03E-1
F9	3.07E-4	9.00E-3	3.07E-4	6.88E-3	3.07E-4	6.29E-5	3.07E-4	1.93E-7
F10	3	7.69E-5	3	1.67E-15	3	6.85	3	3.41E-15

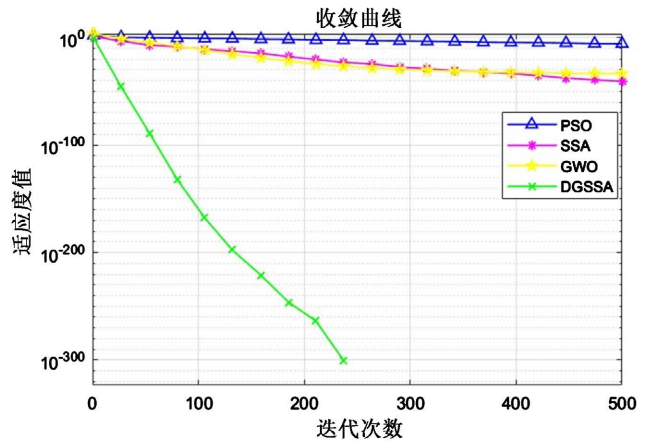
5. 结论

本文改进了麻雀优化算法，解决了其在收敛速度、寻优精度和局部最优问题上的不足。

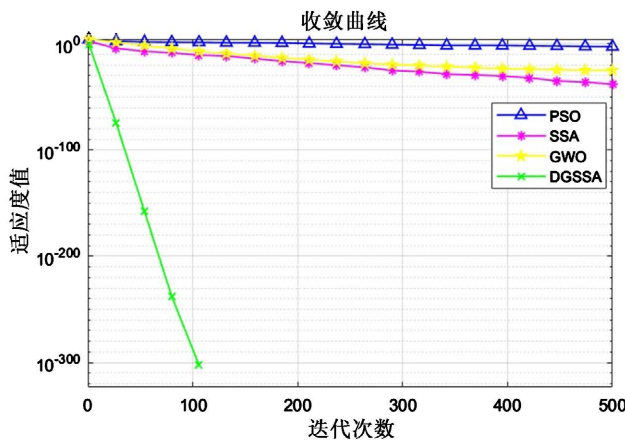
改进的麻雀优化算法利用动态反向学习策略进行种群初始化，增加种群的初始多样性，为搜索奠定了良好的基础；引入黄金正弦因子快速扫描潜在的优化区域，既提高了收敛速度，又平衡了全局和局部



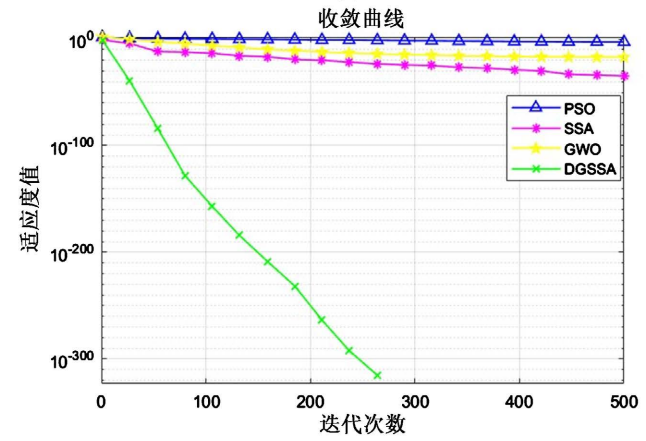
(a) F1



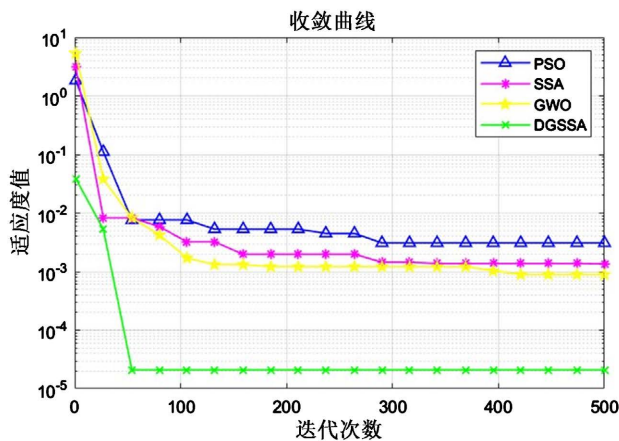
(a) F2



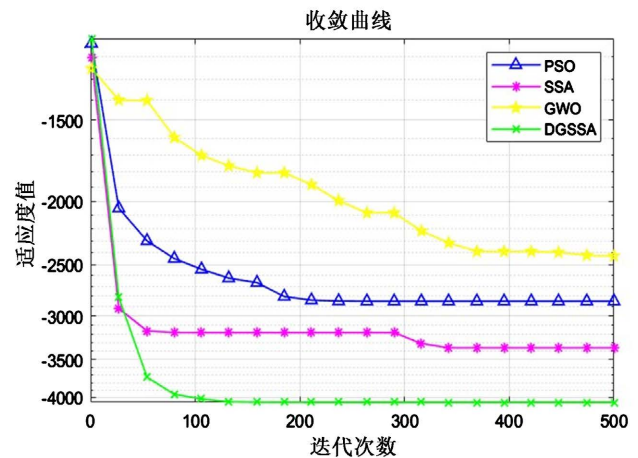
(c) F3



(d) F4



(e) F5



(f) F6

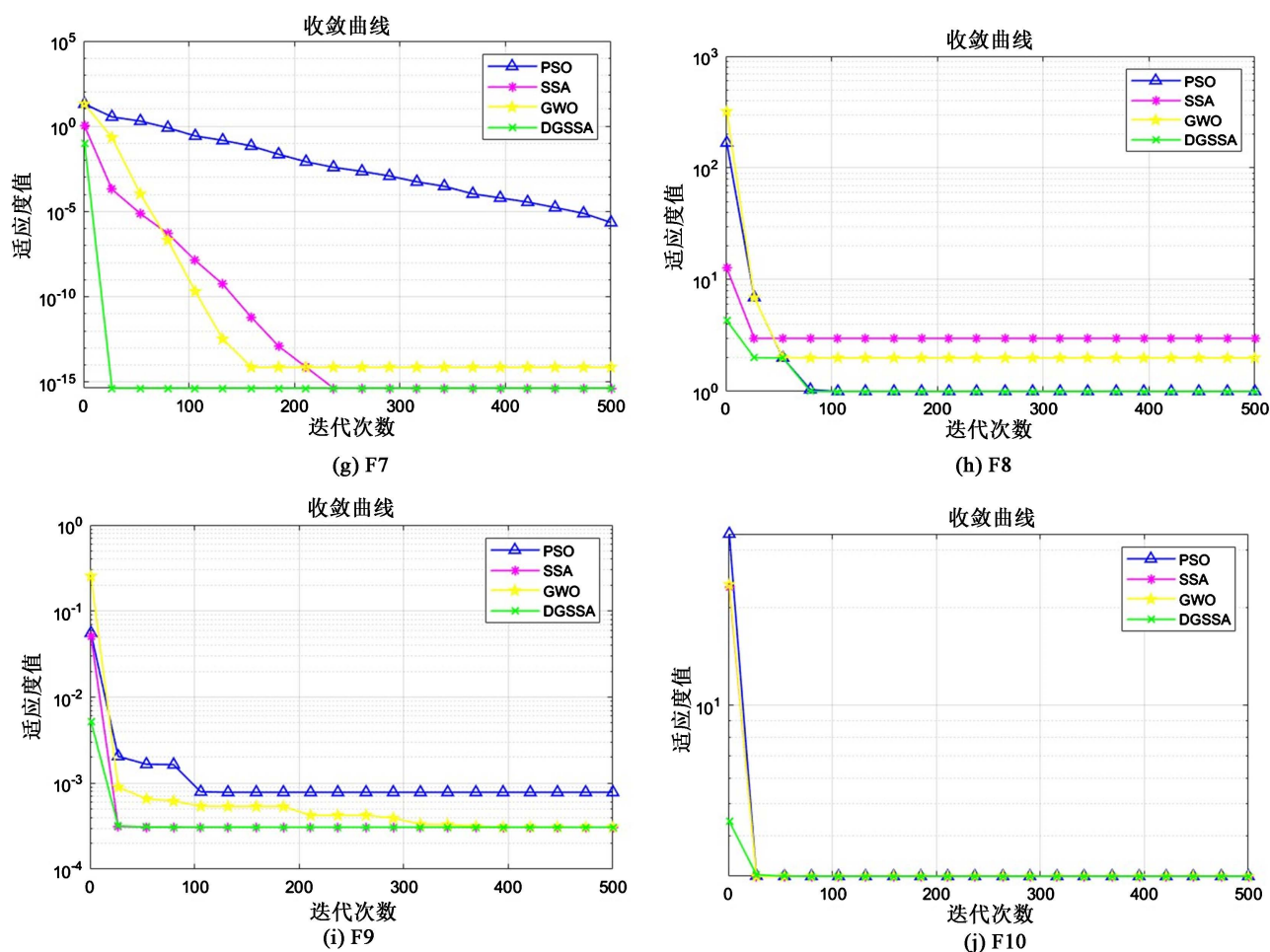


Figure 2. Convergence curve
图 2. 收敛曲线图

搜索的能力；通过多项式变异扰动以及双面镜理论边界优化，能够明显地增加物种多样性，有效突破局部最优解。实验验证表明，相较于传统的麻雀优化算法、灰狼优化算法和粒子群优化算法，改进的麻雀优化算法快速收敛到全局最优解，有更好的收敛精度和更快的收敛速度，提高了算法的实际应用性和效率。

基金项目

大学生创新创业训练计划资助项目(2023569, 2023039, 2023591)。

参考文献

- [1] Xue, J. and Shen, B. (2020) A Novel Swarm Intelligence Optimization Approach: Sparrow Search Algorithm. *Systems Science and Control Engineering*, **8**, 22-34. <https://doi.org/10.1080/21642583.2019.1708830>
- [2] Mirjalili, S., Mirjalili, S.M. and Lewis, A. (2014) Grey Wolf Optimizer. *Advances in Engineering Software*, **69**, 46-61. <https://doi.org/10.1016/j.advengsoft.2013.12.007>
- [3] Kennedy, J. and Eberhart, R. (1995) Particle Swarm Optimization. *Proceedings of ICNN'95-International Conference on Neural Networks*, Perth, 27 November 1995 - 1 December 1995, 1942-1948. <https://doi.org/10.1109/ICNN.1995.488968>
- [4] Mirjalili, S. and Lewis, A. (2016) The Whale Optimization Algorithm. *Advances in Engineering Software*, **95**, 51-67. <https://doi.org/10.1016/j.advengsoft.2016.01.008>

-
- [5] Karaboga, D. and Basturk, B. (2007) Artificial Bee Colony (ABC) Optimization Algorithm for Solving Constrained Optimization Problems. In: Melin, P., Castillo, O., Aguilar, L.T., Kacprzyk, J. and Pedrycz, W., Eds., *Foundations of Fuzzy Logic and Soft Computing*. IFSA 2007. Springer, Berlin, 789-798. https://doi.org/10.1007/978-3-540-72950-1_77
- [6] Heidari, A.A., Mirjalili, S., Faris, H., *et al.* (2019) Harris Hawks Optimization: Algorithm and Applications. *Future Generation Computer Systems*, **97**, 849-872. <https://doi.org/10.1016/j.future.2019.02.028>
- [7] Trojovská, E., Dehghani, M. and Trojovský, P. (2022) Zebra Optimization Algorithm: A New Bio-Inspired Optimization Algorithm for Solving Optimization Algorithm. *IEEE Access*, **10**, 49445-49473. <https://doi.org/10.1109/ACCESS.2022.3172789>
- [8] Zolf, K. (2023) Gold Rush Optimizer: A New Population-Based Metaheuristic Algorithm. *Operations Research and Decisions*, **33**, 113-150. <https://doi.org/10.37190/ord230108>
- [9] 杜云, 周志奇, 贾科进, 等. 混合多项自适应权重的混沌麻雀搜索算法[J/OL]. 计算机工程与应用: 1-15. <http://kns.cnki.net/kcms/detail/11.2127.TP.20230923.0706.006.html>, 2023-12-15.
- [10] 张恩辅, 段冰冰, 刘津平, 等. 基于改进麻雀搜索算法的优化型极限学习机[J]. 软件工程, 2023, 26(9): 18-24.
- [11] Ouyang, C., Qiu, Y. and Zhu, D. (2021) Adaptive Spiral Flying Sparrow Search Algorithm. *Scientific Programming*, **2021**, Article ID: 6505253. <https://doi.org/10.1155/2021/6505253>
- [12] Xu, Y., Yang, Z., Li, X., *et al.* (2020) Dynamic Opposite Learning Enhanced Teaching—Learning-Based Optimization. *Knowledge-Based Systems*, **188**, Article ID: 104966. <https://doi.org/10.1016/j.knosys.2019.104966>
- [13] Tanyildizi, E., and Demir, G. (2017) Golden Sine Algorithm: Anovel Math-Inspired Algorithm. *Advances in Electrical and Computer Engineering*, **17**, 71-78. <https://doi.org/10.4316/AECE.2017.02010>
- [14] 黄清宝, 李俊兴, 宋春宁, 等. 基于余弦控制因子和多项式变异的鲸鱼优化算法[J]. 控制与决策, 2020, 35(3): 559-568.
- [15] Wang, W., Li, K., Tao, X., *et al.* (2020) An Improved MOEA/D Algorithm with an Adaptive Evolutionary Strategy. *Information Sciences*, **539**, 1-15. <https://doi.org/10.1016/j.ins.2020.05.082>