

文章编号: 2095-2163(2023)04-0078-07

中图分类号: TP18

文献标志码: A

基于旋转状态转移的麻雀搜索算法

刘西全, 周新志

(四川大学 电子信息学院, 成都 610065)

摘要: 针对现有的麻雀搜索算法(SSA)在接近全局最优时,种群多样性减少,容易陷入局部最优等问题,提出一种精英旋转策略的麻雀搜索优化算法(RLSSA)。首先,对麻雀种群中发现者进行旋转状态转移变换,以提高算法的局部搜索能力,提高算法的搜索精度。其次,对加入者引入基于莱维飞行的搜索策略,提高算法的全局搜索能力,同时引入贪婪算法搜索寻优策略。最后,在12个基础函数上进行仿真实验。实验结果表明上述方法在收敛速度和精度上有显著优势。

关键词: 麻雀搜索算法; 旋转状态转移; 莱维飞行; 贪婪寻优策略

Sparrow search algorithm based on rotational state transition

LIU Xiquan, ZHOU Xinzhi

(College of Electronics and Information Engineering, Sichuan University, Chengdu 610065, China)

【Abstract】 Aiming at the problems that the existing sparrow search algorithm (SSA) is close to the global optimum, the population diversity decreases and it is easy to fall into the local optimum, a sparrow search optimization algorithm (RLSSA) with elite rotation strategy is proposed. Firstly, the rotational state transition transformation of the discoverer in the sparrow population is carried out to improve the local search ability of the algorithm and improve the search accuracy of the algorithm. Secondly, the search strategy based on Levy flight is introduced to the joiners, which improves the global search ability of the algorithm, meanwhile introduces the greedy algorithm search optimization strategy. Finally, simulation experiments are carried out on 12 basic functions. Experimental results show that the above method has significant advantages in convergence speed and accuracy.

【Key words】 sparrow search algorithm; rotational state transition; Levy flight; greedy optimization strategy

0 引言

近年来,群体智能优化算法在智能机器学习上被广泛应用,群体智能优化算法主要是模仿昆虫、动物、鱼类、鸟类的群体行为、根据其觅食行为启发而提出的算法^[1]。如粒子群优化算法^[2](PSO)、蚁群优化算法^[3](ACO)、灰狼优化算法^[4](GWO)、天鹰优化器^[5](AO)和麻雀搜索算法^[6](SSA)等。其中,麻雀搜索算法是在2020年由薛建凯提出的一种新型智能优化算法,是通过模拟麻雀在自然界的捕食行为来进行研究与开发的。与其他智能优化算法相比,SSA具有调节参数少、搜索精度高、收敛速度快、稳定性强等优势。然而,在SSA收敛过程中,却存在容易陷入局部最优、收敛停滞等问题,直接影响算法的优化效果。

为了提高麻雀优化算法(SSA)的性能,国内外学者提出了许多改进方法。研究发现初始种群对寻优结果有至关重要的作用。为了优化初始种群,吕鑫等学者^[7]、Liu等学者^[8]分别提出tent混沌映射和改进Circle混沌映射对初始种群进行优化。钱敏等学者^[9]在麻雀初始化种群个体位置时引入反向学习策略,以提高寻优前期麻雀种群整体质量,加快全局搜索速度。Ouyang等学者^[10]提出一种自适应搜索策略以提高局部搜索能力,避免种群陷入局部最优。Ma等学者^[11]在文章中引用三角相似理论设计了相似扰动函数,提高算法的搜索能力。Liang等学者^[12]对算法的边界约束进行改进以提高算法的收敛精度。毛清华等学者^[13]通过引入柯西变异算子,来增强算法跳出局部最优空间的能力。

基金项目: 中国民用航空局民航联合研究基金(U1933123)。

作者简介: 刘西全(1995-),男,硕士研究生,主要研究方向:智能信息处理;周新志(1966-),男,博士,教授,主要研究方向:信息感知与机器学习、智能优化决策与智能系统等。

通讯作者: 周新志 Email: xz.zhou@scu.edu.cn

收稿日期: 2022-11-02

1 麻雀搜索算法

在SSA中,麻雀种群可分为3类,分别为:发现者、加入者、警戒者。其中,发现者为整个种群提供丰富食物的方向与位置;跟随者是根据发现者提供的食物方向与位置进行食物的搜索;警戒者是为整个种群提供警戒侦察,一旦发现捕食者,群体边缘麻雀则向安全区域移动获取食物。

(1)发现者。在SSA中,发现者往往根据其自身的更高适应度发挥领导作用,其位置更新策略如下:

$$X_i^{t+1} = \begin{cases} X_i^t \cdot \exp\left(\frac{-i}{\alpha \cdot M}\right) & \text{if } R_2 < ST \\ X_i^t + Q \cdot L & \text{if } R_2 > ST \end{cases} \quad (1)$$

其中, X_i^t 表示第 t 代第 i 个麻雀的位置; t 是当前算法的迭代次数; M 是最大迭代次数; α 是 $(0,1]$ 之间的均匀随机数; L 是大小为 $1 * d$ 的矩阵; Q 是服从标准正态分布的随机数; R_2 和 ST 分别为警戒值和安全值。当 $R_2 < ST$ 时,表示麻雀目前的觅食行为中没有出现捕食者,可以在周围加大搜索范围,当 $R_2 > ST$ 时,表示出现捕食者,需要离开此收缩范围,并对同伴进行报警。

(2)加入者。加入者跟随发现者,一旦其发现其寻到更好的食物,则会离开现在位置去争夺食物,其更新策略如下:

$$X_i^{t+1} = \begin{cases} Q \cdot \exp\left(\frac{X_{worst}^t - X_i^t}{i^2}\right) & \text{if } i > \frac{n}{2} \\ X_p^{t+1} + |X_i^t - X_p^{t+1}| \cdot A^+ \cdot L & \text{otherwise} \end{cases} \quad (2)$$

其中, X_{worst}^t 表示第 t 代时麻雀的最差位置; X_p^{t+1} 是 $t+1$ 代时麻雀的最优位置; 矩阵 A 是 $1 \times d$ 的矩阵,其值为 1 或 -1。如果 $i > \frac{n}{2}$, 表示第 i 个麻雀在此搜索范围内获取食物失败,需要前往其他地方觅食;反之,麻雀继续在此范围内觅食。

(3)警戒者。警戒者为麻雀种群提供危险预知,其更新策略如下:

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

其中, X_{best}^t 表示第 t 代麻雀的最优值; β 是随机的步长因子,服从标准正态分布; K 表示麻雀的运动方向,是 $[-1,1]$ 的随机数; f_g 和 f_w 分别是最优和最差适应度值; ε 是一个极小的常数,防止分母出现

零值。若 $f_i > f_g$, 则表示警戒麻雀在麻雀搜索种群的边缘,随时受到捕食者的袭击,因此需要前往麻雀种群范围内觅食,反之则表示警戒者已经靠近安全麻雀种群内觅食。

2 改进的麻雀搜索算法

2.1 精英旋转搜索策略

状态转移算法^[14] (State Transition Algorithm, STA)是由周晓君等学者于2012年提出的一种全局优化算法,包括旋转、平移、伸缩、坐标四种状态转移的变换,旋转状态转移变换的数学模型如下:

$$s_{k+1} = s_k + \alpha \cdot \frac{1}{n \cdot \|s_k\|_2} \cdot R_r \cdot s_k \quad (4)$$

其中, n 表示计算维度; α 为旋转因子,表示在半径为 α 的半径范围内寻找下一组最优解; R_r 是 $[-1,1]$ 的随机数; $\|\cdot\|$ 表示向量的二范数。

由于麻雀发现者在新一代的探索中是随机状态,脱离上一代的最优种群,容易陷入局部最优。因此,本文对精英麻雀发现者进行旋转状态转移变换,增加麻雀的种群多样性,经状态转移后的新种群为:

$$\text{new } X_i^{t+1} = X_{best}^t + \alpha \cdot \frac{1}{d \cdot \|X_{best}^t\|_2} \cdot R_r \cdot X_{best}^t \quad (5)$$

2.2 融合莱维飞行的搜索策略

莱维飞行是服从莱维分布的一种随机搜索方法,是一种短距离的搜索与偶尔较长距离的行走相间的行走方式,从而促使莱维飞行具有良好的全局搜索能力,莱维飞行的位置更新公式为:

$$X_i^{t+1} = X_i^t + \delta \oplus \text{Levy}(\beta) \quad (6)$$

其中, δ 表示步长大小, $\delta > 0$, 通常设置为 $\delta = 1$; “ \oplus ” 表示点对点乘法; $\text{Levy}(\beta)$ 表示服从莱维分布的路径,并且满足:

$$\text{Levy} \sim u = t^{-\beta} \quad 1 < \beta \leq 3 \quad (7)$$

使用 Mantegna 方法来生成莱维飞行的随机步长 s 为:

$$s = \frac{u}{|v|^{\frac{1}{\beta}}} \quad (8)$$

$$u \sim N(0, \delta_u^2), v \sim N(0, \delta_v^2) \quad (9)$$

$$\delta_u = \left\{ \frac{\Gamma(1+\beta) \sin \pi \beta / 2}{\Gamma[(1+\beta)/2] \beta \cdot 2^{(\beta-1)/2}} \right\}^{\frac{1}{\beta}}, \delta_v = 1 \quad (10)$$

在这里, β 设置为 1.95。

本文采用莱维飞行搜索策略进行全局探测,使得全部麻雀个体广泛分布在整个搜索空间中,提高全体

麻雀的全局搜索能力。经过对麻雀加入者位置适应度值较优的个体进行莱维扰动后,得到的新种群为:

$$new X_i^{t+1} = X_i^t(1 + s) \quad (11)$$

2.3 贪婪算法寻优

虽然旋转状态转移和莱维飞行可以实现位置的更新,但是其无法保证得到的新解较原解更优,因此对新解和原解的适应度值进行比较,通过贪婪算法机制来保留最优的适应度值和解。具体公式如下:

$$X_i^t = \begin{cases} X_i^t & fit(new X_i^t) > fit(X_i^t) \\ new X_i^t & fit(new X_i^t) < fit(X_i^t) \end{cases} \quad (12)$$

2.4 RLSSA 算法流程

RLSSA 算法引入旋转状态转移和莱维分布,增加了种群的多样性,提高算法搜索精度、收敛速度,算法实现流程见如下:

(1)初始化种群。参数主要包括:麻雀的种群规模 N ,迭代次数 M ,发现者和警戒者的比例,目标函数 fit ,麻雀算法的搜索边界范围 Lb 和 Ub 。

(2)初始化适应度值,并排序,得出最优适应度值

f_g 和最差适应度值 f_w 及其各自的种群位置 X_{best}^t 、 X_{worst}^t 。

(3)根据式(1)更新发现者的位置,根据式(5)得到经旋转策略变换的位置。

(4)根据式(2)更新加入者位置,根据式(11)得到莱维飞行扰动的位置。

(5)根据式(3)更新警戒者位置。

(6)一次迭代结束,计算适应度值,并根据式(12)选择新一代的适应度值和麻雀位置。

(7)根据当前麻雀种群状态,计算最优适应度值 f_g 和最差适应度值 f_w 及其各自的种群位置 X_{best}^t 、 X_{worst}^t 。

(8)判断算法是否满足最大迭代次数,满足则退出,输出结果,否则重复执行步骤3~7。

3 实验结果及分析

3.1 实验设计

为了验证 RLSSA 的有效性和优越性,采用基准测试函数进行仿真实验,见表1。

表1 基准测试函数

Tab. 1 Benchmark functions

类型	测试函数	维度	搜索范围	最优值
单峰测试函数	$F_1 = \sum_{i=1}^n x_i^2$	30	$[-100, 100]$	0
	$F_2 = \sum_{i=1}^n x_i + \prod_{i=1}^n x_i $	30	$[-10, 10]$	0
	$F_3 = \sum_{i=1}^n [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	30	$[-30, 30]$	0
	$F_4 = \sum_{i=1}^n (x_i + 0.5)^2$	30	$[-100, 100]$	0
多峰测试函数	$F_5 = \sum_{i=1}^n -x_i \sin \sqrt{ x_i }$	30	$[-500, 500]$	-12 569.487
	$F_6 = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i)] + 10$	30	$[-5.12, 5.12]$	0
	$F_7 = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}\right) - \exp\left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)\right) + 20 + e$	30	$[-32, 32]$	0
固定多峰测试函数	$F_8 = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	30	$[-600, 600]$	0
	$F_9 = \left(\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^2 (x_i - a_{ij})^6}\right)^{-1}$	2	$[-65, 65]$	1
	$F_{10} = \sum_{i=1}^{11} \left[a_i - \frac{x_1(b_i^2 + b_1 x_2)}{b_i^2 + b_1 x_3 + x_4} \right]^2$	4	$[-5, 5]$	$3.07 \times e^{-4}$
	$F_{11} = -\sum_{i=1}^4 c_i \exp\left(-\sum_{j=1}^6 a_{ij} (x_j - P_{ij})^2\right)$	6	$[0, 1]$	-3.32
	$F_{12} = \sum_{i=1}^{10} [(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	$[0, 10]$	-10.536 3

表 1 中, $F_1 \sim F_4$ 为单峰测试函数, 用来测试函数的收敛速度和精度; $F_5 \sim F_8$ 为多峰测试函数, 主要用来测试算法跳出局部最优的能力; $F_9 \sim F_{12}$ 为固定维多峰测试函数, 这类函数主要表现为工程优化问题。为了验证算法的有效性和优越性, 将本文

改进后的旋转麻雀算法 (RLSSA) 与 GWO、WOA、PSO、SSA 及一种基于相似变换改进的麻雀搜索算法 EMSSA 进行对比求解实验。

图 1 为部分函数的三维图, 依次分别为 F_1 、 F_2 、 F_5 、 F_8 、 F_9 、 F_{12} 。

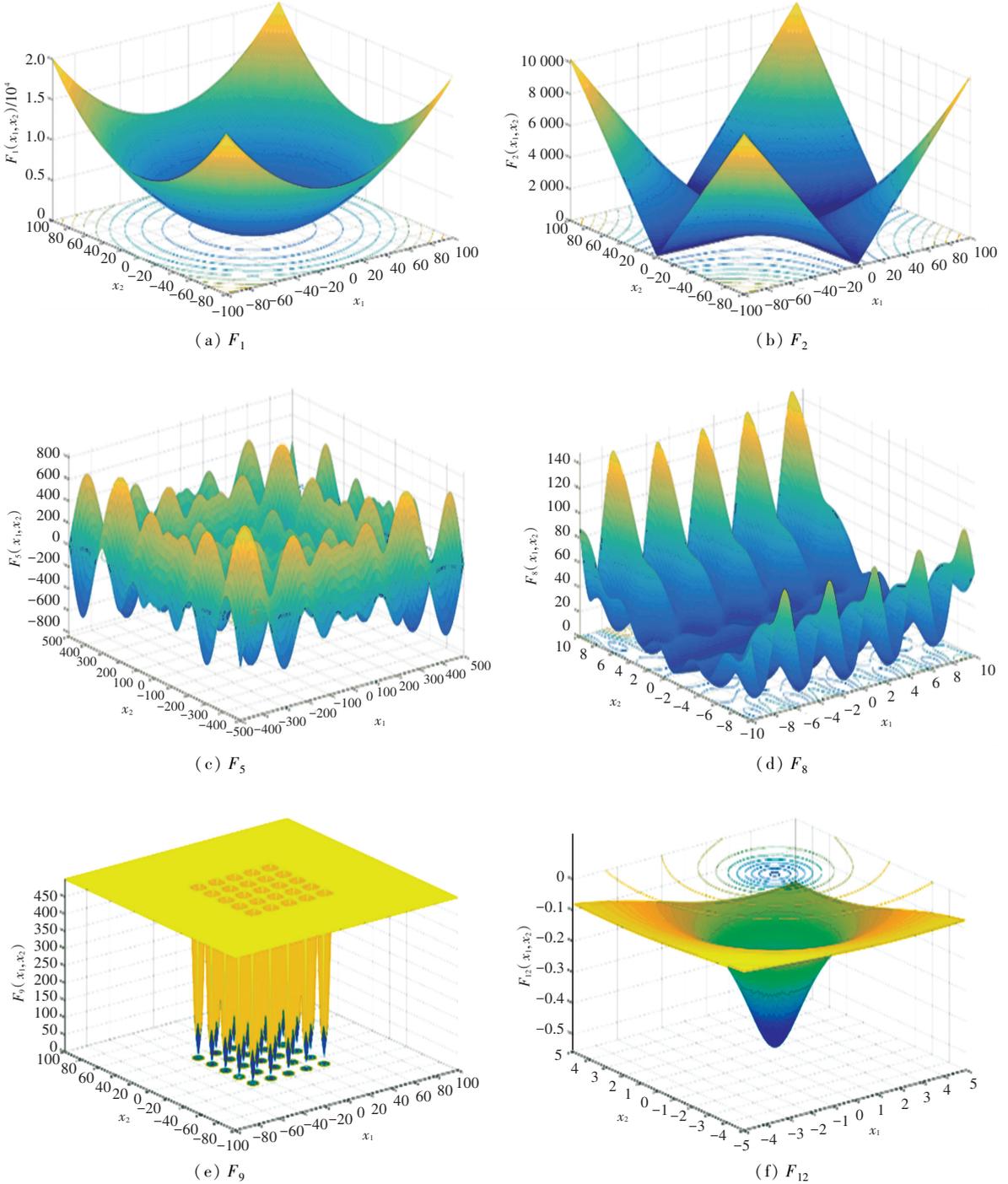


图 1 函数三维图

Fig. 1 Three-dimensional diagram of functions

3.2 仿真实验结果及分析

本实验是在 Windows10 系统, Intel (R) Core

(TM) i5-62--U CPU@ 2.30 GHz, 内存 8.00 GB, Matlab R2016a 的环境下进行的, 实验种群规模 $N =$

30,最大迭代次数 $M = 500$,发现者和警戒者的比例分别为20%和10%,每个目标函数的维度 d 和边界 Lb 和 Ub 由表1中给出。考虑到每次算法寻优都具有偶然性,所以对每个基准函数都运行30次寻优运

算,由此计算出的平均值和标准差作为最终的评价指标。表2和表3分别给出了12个基准测试函数在GWO、WOA、PSO、SSA、EMSSA、RLSSA优化算法后的均值和方差。

表2 基准测试函数优化结果均值比较

Tab. 2 Comparison of the mean of the optimization results of the benchmark functions

函数	GWO	WOA	PSO	SSA	EMSSA	RLSSA
F_1	9.962E-28	9.861E-74	0.000 155 45	1.589E-69	0	1.57E-301
F_2	7.918E-17	6.166E-52	0.034 572 45	8.049E-31	0	0
F_3	26.916 274 60	28.097 178 30	95.667 894 20	2.887E-05	0.000 124 78	2.437E-05
F_4	0.938 244 72	0.454 720 52	0.000 212 49	2.203E-11	3.020E-07	7.181E-09
F_5	-6 213.364	-10 468.755	-4 663.012 1	-8 650.574 7	-12 569.486	-9 093.657 6
F_6	2.700 641 70	0	60.831 647 40	0	0	0
F_7	1.084E-13	3.806E-15	0.088 039 77	8.881E-16	8.881E-16	8.881E-16
F_8	0.045 882 40	0.023 273 75	2.216E-06	7.256E-13	1.706E-08	1.293E-09
F_9	4.117 671 50	1.843 119 17	2.761 966 23	4.687 244 72	8.542 262 73	7.463 712 17
F_{10}	0.003 972 00	0.000 791 06	0.000 859 67	0.000 319 62	0.001 665 21	0.000 334 67
F_{11}	-3.262 124 00	-3.251 718 00	-3.283 779 00	-3.258 302 00	-2.549 268 00	-3.275 134 10
F_{12}	-10.244 672 00	-7.249 665 40	-10.069 509 00	-9.100 676 90	-10.536 232 00	-10.536 219 00

表3 基准测试函数优化结果方差比较

Tab. 3 Comparison of variance of the optimization results of the benchmark functions

函数	GWO	WOA	PSO	SSA	EMSSA	RLSSA
F_1	3.054E-27	8.496E-72	0.000 142 8	1.308E-63	0	0
F_2	4.765E-17	2.685E-51	0.086 721 6	8.812E-29	0	2.16E-133
F_3	0.692 666 2	0.499 712 7	92.289 922	7.236E-05	0.000 204 4	0.000 213 2
F_4	0.692 666 2	0.499 712 7	92.289 922	7.236E-05	0.000 204 4	0.000 213 2
F_5	866.728 28	1 681.926 1	1 475.411 2	654.083 28	431.691 13	1 846.946 4
F_6	3.584 067 4	2.075E-14	15.258 185 0	0	0	0
F_7	1.750E-14	2.962E-15	0.383 702 9	1.002E-31	1.002E-31	1.002E-31
F_8	0.006 411 0	0.022 910 4	0.014 355 4	0	0	0
F_9	3.783 192 0	2.484 884 0	2.499 225 6	5.275 406 6	5.235 415 2	5.735 046 5
F_{10}	0.008 121 0	0.002 124 2	0.000 146 4	5.800E-05	3.029E-05	2.80E-05
F_{11}	0.077 856 6	0.114 990 1	0.057 004 8	0.060 328 3	0.574 797 3	0.028 314 1
F_{12}	2.073 365 9	3.471 107 5	2.144 166 4	2.421 391 0	0.000 296 4	0.000 206 3

由表2和表3可知,对于高维单峰函数 $F_1 \sim F_4$,RLSSA较其他5种算法有较小的适应度值和较小的标准差,说明RLSSA有较高的寻优精度和寻优稳定性。在多峰测试函数 $F_5 \sim F_8$ 上, F_5 函数在EMSSA上有最好的精度,RLSSA对比其他4种算法有较优的精准度。而在 $F_6 \sim F_8$ 上,RLSSA能跳出局部最优,稳定找到全局最优解。对固定维多峰测试函数 $F_9 \sim F_{12}$,RLSSA较其他5种算法的均值和标准差提升并不高,但是在寻优精度和稳定性上都

要优于其他5种算法。

3.3 收敛曲线分析

为了反映出RLSSA的收敛速度,在500次迭代中画出各个算法的收敛特性曲线,如图2所示。在高维单峰函数 $F_1 \sim F_4$ 的曲线中可以看出,RLSSA的收敛速度均优于其他算法,同时还看出,RLSSA能快速跳出算法的局部最优解。在高维多峰函数 F_5 的曲线中看出RLSSA的精度不如EMSSA和WOA算法,比其他3种算法更优,这是因为莱维飞

行策略是一种较短步长多, 较长步长少的策略, 算法不能通过合适的步长实现全局寻优。在 $F_6 \sim F_8$ 中, RLSSA 收敛速度快、精度高; 在固定多峰函数 $F_9 \sim$

F_{12} 的曲线中, 算法结果都趋于稳定, 并且结果相近, 但可以看出 RLSSA 的收敛速度或者精度优于其他算法。

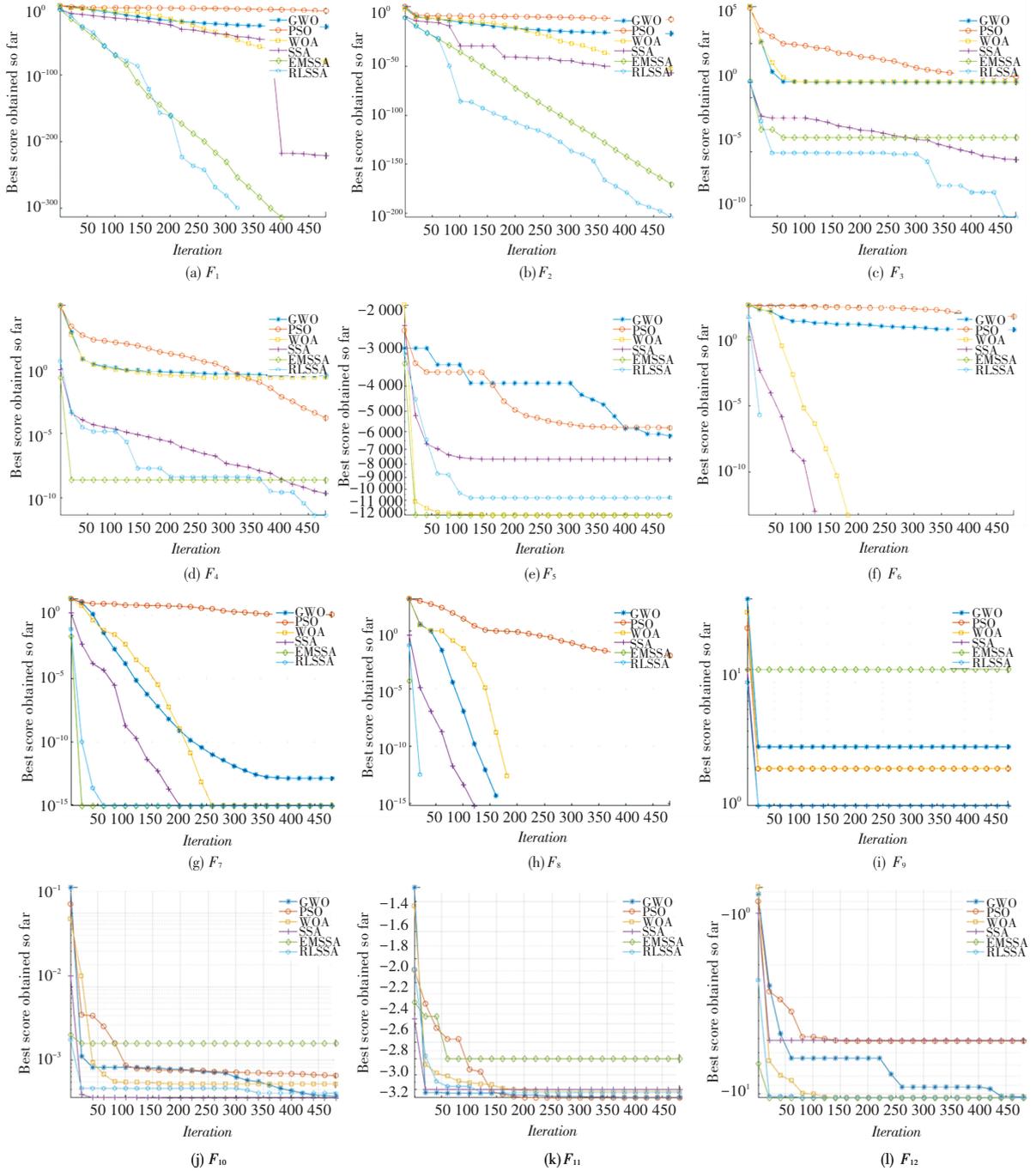


图 2 基准测试函数的收敛曲线

Fig. 2 Convergence curve of the benchmark functions

4 结束语

针对传统麻雀搜索算法 (SSA) 容易陷入局部最优、收敛停滞等问题, 提出了一种融合旋转状态转移的麻雀搜索算法 (RLSSA), 改进的算法具有收敛速度快、剪度高、剪定性强等优势。首先, 对于麻雀发

现者, 引入一种旋转状态转移策略的搜索方法以提高算法跳出局部最优的能力; 其次, 引入莱维飞行策略对麻雀加入者加以改进, 此方法可以提高麻雀算法的全局搜索能力; 最后, 通过贪婪算法的准则, 对原算法和改进的算法进行择优。通过 12 个基准测 (下转第 90 页)