

双核因素蝙蝠算法

肖海军^{1,2}, 成金华², 何凡^{1,*}

(1 中国地质大学(武汉) 数学与物理学院, 武汉 430074;

2 中国地质大学(武汉) 经济管理学院 资源环境经济研究中心, 武汉 430074)

摘要 提出了一种基于蝙蝠算法的新型仿生优化算法——双核因素蝙蝠算法(DCFBA)。通过改变蝙蝠算法中的速度更新公式,可有效提高寻优效率。为了验证 DCFBA 的性能,在 9 个测试函数上使用标准蝙蝠算法(BA)、粒子群算法(PSO)和 DCFBA 进行了实验。结果表明:DCFBA 在算法的有效性、优越性和稳定性上都优于 BA 和 PSO 算法。

关键词 蝙蝠算法;优化算法;双核因素蝙蝠算法;速度更新公式

中图分类号 TP273 **文献标识码** A **文章编号** 1672-4321(2018)01-0132-06

Double Core Factors Bat Algorithm

Xiao Haijun^{1,2}, Cheng Jinhua², He Fan¹

(1 School of Mathematics and Physics, China University of Geosciences, Wuhan 430074, China;

2 Research Center of Resources and Environmental Economics, China University of Geosciences, Wuhan 430074, China)

Abstract In this paper, a new bionic optimization algorithm which termed double core factors bat algorithm (DCFBA) based on the bat algorithm was proposed. A new velocity updating formula was proposed in our algorithm to improve the optimization efficiency. Aimed at showing the advantages of our new algorithm, 9 benchmark problems were performed by the classical bat algorithm, the particle swarm optimization and DCFBA respectively. Experimental results show that DCFBA is better than the classical bat algorithm and the particle swarm optimization in effectiveness, superiority and stability.

Keywords bat algorithm; optimization algorithm; double core factors bat algorithm; velocity updating formula

1 研究背景

通过对生物行为的模拟,人们发现了均衡性强、优化效果好的群智能优化算法。受到群智能优化算法的启发,研究人员又提出了广泛应用的智能优化算法。如:随机扩散搜索^[1],蚁群优化^[2],粒子群优化^[3-4]和蝙蝠算法^[5]。Bishop^[1,6]提出的随机扩散搜索(SDS)是一种具有良好数学概率的全局搜索算法^[7-9],虽然在全局搜索过程中 SDS 效率高、鲁棒性好,但其优化技术只适用于可分解为多个部分函数的目标函数。蚁群优化算法^[2](ACO)通过模拟“蚂蚁”记录位置和生物特征来寻找最优路径,这使得

蚂蚁在后续的探索中找到更优的路径^[10]。但是 ACO 的最优解存在不确定性。粒子群优化算法(PSO)可以在 n 维空间中搜寻点或面的最优解^[11-13],该算法处理局部极值问题时灵活性较强。但是 PSO 算法需要大量的粒子群个体才能获得相对满意的优化结果。蝙蝠算法(BA)是 Yang^[5]提出的一种全局优化算法(蝙蝠通过模拟脉冲发射频率和响度来搜索猎物)通过调整频率来加强解的多样性,并利用自动缩放能力来平衡寻优过程中的搜索与开发。

与其他算法相比,蝙蝠算法在处理一些复杂问题时表现更好。因此,一些研究人员有效地改进了蝙蝠算法。如:KHan 等^[14]提出了模糊逻辑蝙蝠算法,将模糊逻辑与蝙蝠算法相结合。Yang^[15]提出了一种

收稿日期 2017-10-08 * 通讯作者 何凡 研究方向:智能计算 E-mail: cughefan@qq.com

作者简介 肖海军(1965-)男 教授 博士 研究方向:数据挖掘 E-mail: xiaohj@cug.edu.cn

基金项目 国家自然科学基金资助项目(11301492)

多目标蝙蝠算法来解决某些工程问题. Lin 等^[16]提出了组合蝙蝠算法, 通过使用 Lévy 飞行和组合映射来估计算法参数. Nakamura 等^[17]提出名为二进制蝙蝠算法的离散版本来改进分类和特征选择. Xie 等^[18]提出了基于差分算子和 Lévy 飞行的蝙蝠算法, 借此来解决优化问题.

此外, 还有各种优化的蝙蝠算法. 例如, Zhang 和 Wang^[19]引入生物突变, 来提高解决图像匹配多样性的问题. Xie 和 Zhou^[20]提出了组合蝙蝠算法, 通过组合蝙蝠算法与回声搜索来优化基准函数的数值. Fister I 等^[21]使用差分进化对蝙蝠路径进行局部搜索, 然后结合四元数创建了四元数蝙蝠算法 (QBA), 这种方法可以解决大规模的旋转型和几何计算型的优化问题^[22]. Li 和 Zhou^[23]提出了一种基于复数编码的新型蝙蝠算法, 它对编码的实部和虚部分别单独更新.

BA 从搜索转向开发时收敛迅速, 当频率、波长或响度变化太快时, 可能导致局部最优. 因此, 本文提出一种改进的蝙蝠算法, 称为双核因素蝙蝠算法 (DCFBA). 在 DCFBA 中, 自适应速度更新公式加入了两个自适应学习因子并且受到最优和次优蝙蝠位置的影响, 再通过仿真实验验证 DCFBA 的有效性.

2 蝙蝠算法

蝙蝠是一种具有回声定位能力的动物. 搜索猎物时, 蝙蝠发出的声波碰到一个物体后, 声波会返回到它的耳中^[13], 蝙蝠可以由此估计与物体的距离^[24]. 基于蝙蝠回声定位的特性与狩猎习性, Yang^[25]提出了标准蝙蝠算法, 标准规则如下.

- (1) 所有蝙蝠通过回声定位感知距离, 并以某种独特的方式从背景障碍中区分猎物;
- (2) 蝙蝠在位置 S_i 以固定的频率 f_{\min} 和速度 v_i , 变化的波长 λ 和响度 A_0 自由地狩猎;
- (3) 假设响度从最大值 A_0 到最小值 A_{\min} 不等.

2.1 蝙蝠的速度更新和位置更新

假设搜索空间维度是 d , 可以根据以下公式更新蝙蝠 i 在时间 t 上的速度 v_i^t 和位置 X_i^t :

$$f_i = f_{\min} + \alpha(f_{\max} - f_{\min}), \quad (1)$$

$$v_i^{t+1} = v_i^t + (X_i^t - X_{best}^t)f_i, \quad (2)$$

$$X_i^{t+1} = X_i^t + V_i^{t+1}, \quad (3)$$

其中 α 代表 $[0, 1]$ 中的随机数, f_i 表示当前时刻蝙蝠 i 的变化频率, X_{best}^t 是当前的全局最优解, 可以通过比较所有解找到.

为了改善解的多样性, 从蝙蝠群体中选择一只蝙蝠, 并根据 (4) 式随机地产生一个新的局部解. 通过此设置, 局部搜索可以顺利进行, 并从已经选择的解中精选新的解.

$$X_{new} = X_{old} + \delta A^t, \quad (4)$$

其中 A^t 表示此时所有蝙蝠的平均响度, $\delta \in [-1, 1]$ 是一个随机值, X_{old} 代表从当前最优解中选择的随机解.

2.2 蝙蝠的响度更新和脉冲发射率更新

当搜索开始时, 为了延长超声波传播的距离, 将响度设置为最大值, 将脉冲发射率设置为最小值. 当蝙蝠搜索猎物时, 响度逐渐降低, 脉冲发射率增大, 这可以帮助蝙蝠更准确地定位猎物. 算法进行迭代时响度 A_i 和脉冲发射率 r_i 的更新公式如下:

$$A_i^{t+1} = \varepsilon A_i^t, \quad (5)$$

$$r_i^{t+1} = r_i^0 [1 - \exp(-\gamma t)], \quad (6)$$

其中 ε 和 γ 是常量, $0 < \varepsilon < 1$, $\gamma > 0$. 很容易知道, 当 $t \rightarrow \infty$ 时, $A_i^t \rightarrow 0$, $r_i^t \rightarrow r_i^0$.

3 双核因素蝙蝠算法的速度更新方法及步骤

速度更新公式在蝙蝠算法中起关键作用, 它决定了蝙蝠在搜索空间的速度. 蝙蝠算法通过 (2) 式更新蝙蝠的速度, 它考虑到以前的速度和当前蝙蝠的最佳位置. 然而, 蝙蝠的最佳位置并不是改变搜索方向的唯一因素. 我们在研究过程中发现蝙蝠的次优位置也是影响优化结果的一个重要因素. 因而在提出的 DCFBA 中我们考虑了这个因素, 这种改进的蝙蝠算法称之为双核因素蝙蝠算法 (DCFBA).

3.1 DCFBA 的速度更新方法

假设蝙蝠群体处于 P 维搜索空间, 蝙蝠 i 在迭代 t 时的速度和位置被描述为向量 $v_i^t = \{v_{i1}^t, v_{i2}^t, \dots, v_{iP}^t\}$ 和 $x_i^t = \{x_{i1}^t, x_{i2}^t, \dots, x_{iP}^t\}$. DCFBA 的速度更新公式定义如下:

$$v_i^{t+1} = v_i^t + f_1 c_1(t) (X_i^t - X_{second}^t) + f_2 c_2(t) (X_i^t - X_{best}^t), \quad (7)$$

其中 X_{best} 和 X_{second} 分别代表其他蝙蝠的最优位置和次优位置, $c_1(t)$ 和 $c_2(t)$ 为学习因子. 更新公式如下:

$$c_1(t) = 1 - \exp(-\Psi(t)), \quad (8)$$

$$c_2(t) = 1 - c_1(t), \quad (9)$$

$$\Psi(t) = |Fitness_{mean}(t) - Fitness_{best}(t)|, \quad (10)$$

$$Fitness_{mean}(t) = \frac{1}{M} \sum_{i=1}^M Fitness(x_i^t). \quad (11)$$

其中 $Fitness_{mean}(t)$ 和 $Fitness_{best}(t)$ 分别表示迭代 t 时蝙蝠群体的平均适应度和最佳适应度值。(7) 式由记忆项和社会认知项构成。记忆项 v_i^{t-1} 表示惯性运动,这决定了蝙蝠在此过程中探索新的搜索区域的能力。在社会认知项中 $c_1(t)$ 和 $c_2(t)$ 分别是(8)和(9)式的两个自适应学习函数。这种动态和非线性变化在早期迭代中有助于更多地偏重于蝙蝠次优经验。在迭代后期 X_{second} 将接近 X_{best} , (7) 式将无限地接近标准蝙蝠算法的速度更新公式。由此可见,BA 是 DCFBA 的特殊形式。

依靠速度更新公式的创新,可以大大加强 DCFBA 在优化过程中的灵活性,使算法具有提高收敛速度的优点,更重要的是可以避免算法陷入局部最优。

3.2 DCFBA 的更新方法

基于上述更新规则,DCFBA 的实施步骤可概括如下。

DCFBA 伪代码

Input:

M : the population size of bats

ε : the loudness

γ : the pulse emission

f_{max} : the maximum pulse frequency

f_{min} : the minimum pulse frequency

$iter_{max}$: the maximum number of iterations

Output:

the bat with the best fitness value in the population.

Initialise the bats' positions X_i , velocity v_i , frequency f_i , pulse rates

r_i and the loudness A_i . Assume the fitness function is minimum.

while the stop condition is not satisfied **do**

for all bats **do**

Calculate the fitness value $Fitness_i$ and obtain the best solution by updating the bats' positions and velocity as described in Equ.(3) and (7).

if $rand > r_i$ then

Choose a solution among the best ones and randomly generate a local solution X_{new} around the best chosen solution.

end if

Evaluate new solution $Fitness_{new}$

if $rand < A_i$ and $Fitness_{new} < Fitness(X_i)$ then

Accept the new solutions, increase r_i and reduce A_i .

end if

Rank the fitness values of all bats and find the best solution X_{best} .

end for

end while

4 数值验证

4.1 基准问题

在实验中,为了研究 DCFBA 的有效性、效率性和稳定性,我们采用 BA 和 PSO 算法作为对比算法,选取了具有不同属性的 9 个基准优化函数。这 9 个基准函数可以分为高维单峰函数(第 I 类)、高维多模态函数(第 II 类)和低维函数(第 III 类),见表 1。

表 1 基准优化函数

Tab.1 Benchmark functions

Class	No.	Name	bounds
I	F_1	Sphere	[-100,100]
	F_2	Problem 1.2	[-100,100]
	F_3	Problem 2.21	[-100,100]
	F_4	Step	[-100,100]
II	F_5	Rastrigin	[-100,100]
	F_6	Ackley	[-32,32]
	F_7	Griewank	[-100,100]
III	F_8	Kowalik	[-5,5]
	F_9	Branin	$x_1 \in [-5,10]$ $x_2 \in [0,15]$

4.2 仿真平台

用于评估 DCFBA 的平台配置为: Intel Pentium (R) 双核 CPU, 2.5GHz, 2.00GB RAM, Windows 7 操作系统, MATLAB 2014a (Windows) 开发环境。

4.3 参数设置

在 DCFBA 算法中蝙蝠群体大小为 20, 响度衰减系数 $\alpha=0.5$, 增加脉冲发射系数 $\gamma=0.5$, 脉冲频率 $f_i \in [0, 2]$. BA 算法的参数与 DCFBA 相同。在 PSO 算法中学习因子 $c_1=c_2=0.5$, 粒子群体大小为 30。

4.4 实验结果

采用 3 种方法在 9 个基准函数上的优化对比结果如表 2~4 所示。其中 Worst, Mean, Best 和 Std. 分别表示最差适应度值, 平均适应度值, 最佳适应度值和适应度值的标准差。黑体数字表示 DCFBA 是优选的, 而加下划线的数值表示其他算法是优选的。

由表 2 可知, DCFBA 可以在第 I 类函数中获得最优结果。对于这 4 个函数, DCFBA 的最差适应度值, 平均适应度值, 最佳适应度值和适应度值的标准差均优于 BA 和 PSO, 这意味着 DCFBA 在高维单峰函数下具有最佳的稳定性。

表 2 函数 F_1, F_2, F_3, F_4 的仿真结果
Tab.2 Simulation results for test functions F_1, F_2, F_3, F_4

Function	Algorithm	Results			Std.
		Worst	Mean	Best	
$F_1(D=30)$	BA	0.002225	0.001659	0.001266	0.000288
	PSO	2.109635	1.370894	0.443358	0.415458
	DCFBA	0.002082	0.001675	0.001202	0.000255
$F_2(D=30)$	BA	3.807914	2.231063	0.876662	0.768364
	PSO	442.921992	237.930082	128.432908	86.359274
	DCFBA	3.188723	1.637706	0.688113	0.742608
$F_3(D=30)$	BA	0.381114	0.227995	0.089119	0.097666
	PSO	0.630338	0.487075	0.273924	0.108376
	DCFBA	0.243295	0.064262	0.021497	0.063461
$F_4(D=30)$	BA	5.000000	3.800000	1.000000	1.469694
	PSO	4.000000	1.700000	0.000000	1.268858
	DCFBA	4.000000	1.500000	0.000000	1.024695

表 3 函数 F_5, F_6, F_7 的仿真结果
Tab.3 Simulation results for test functions F_5, F_6, F_7

Function	Algorithm	Results			Std.
		Worst	Mean	Best	
$F_5(D=30)$	BA	41.133576	31.072079	16.244360	8.254371
	PSO	123.295382	101.450532	67.089144	17.734510
	DCFBA	41.219667	23.621985	12.199450	9.539095
$F_6(D=30)$	BA	3.225116	2.245354	1.648462	0.419659
	PSO	2.528644	2.134810	1.512222	0.275079
	DCFBA	2.411129	1.609800	1.343029	0.339560
$F_7(D=30)$	BA	0.000121	0.000086	0.000071	0.000014
	PSO	0.079945	0.051904	0.021045	0.019340
	DCFBA	0.000103	0.000090	0.000061	0.000012

由表 3 可知,DCFBA 在第 II 类函数中的优化结果也是最好的.即便 DCFBA 在某项指标上不如其他算法,但其结果也是非常接近最优值的.总体而言,

DCFBA 在高维多峰函数上的优化性能优于 BA 和 PSO.

表 4 函数 F_8, F_9 的仿真结果
Tab.4 Simulation results for test functions F_8, F_9

Function	Algorithm	Results			Std.
		Worst	Mean	Best	
$F_8(D=4)$	BA	0.000855	0.000453	0.000308	0.000157
	PSO	0.001594	0.000641	0.000308	0.000513
	DCFBA	0.001173	0.000482	0.000308	0.000309
$F_9(D=2)$	BA	0.768968	0.434996	0.397887	0.111324
	PSO	0.397892	0.397888	0.397887	0.000001
	DCFBA	0.397889	0.397888	0.397887	0.000000

由表 4 可知,对于函数 F_8, F_9 3 种算法的最优适应度值相等,虽然 DCFBA 在平均适应度值和标准偏差上比 BA 稍差,但是 DCFBA 的其他指标都比 BA 和 PSO 好.对于 F_9 ,DCFBA 优于(超过或等于)BA 和 PSO 的结果.显然,在第 III 类函数上,DCFBA 的性能仍然优于 BA 和 PSO.

数 F_1, F_2, F_3, F_5, F_7 和 F_8 的迭代次数从 10 变化到 810.图 4 对应的函数 F_4 迭代次数为 1110,图 9 对应的函数 F_9 迭代次数为 325.函数 $F_1 \sim F_9$ 的维度 D 设置为 30,蝙蝠群体 m 为 20,搜索步长为 0.01.算法经过十折交叉验证后的结果表明:DCFBA 在所有基准函数上的优化结果优于 BA 和 PSO 算法.

针对三类函数,图 1~9 是适应性演化曲线,函

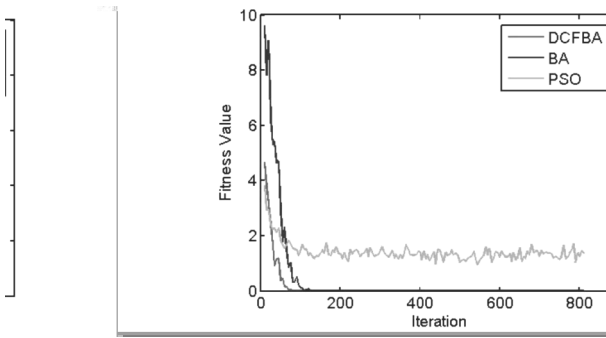


图 1 F_1 适应度值演化曲线 搜索范围为 $[-100, 100]$

Fig.1 Evolution curves of fitness value for F_1 , bounds $\in [-100, 100]$

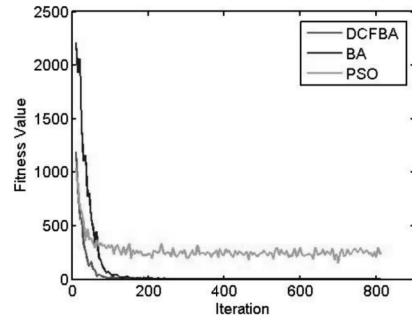


图 2 F_2 适应度值演化曲线 搜索范围为 $[-100, 100]$

Fig.2 Evolution curves of fitness value for F_2 , bounds $\in [-100, 100]$

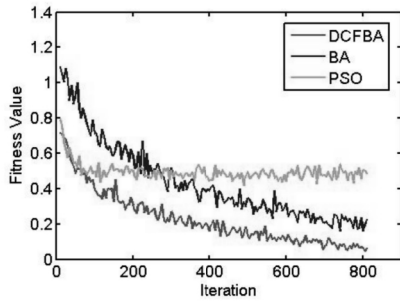


图 3 F_3 适应度值演化曲线 搜索范围为 $[-100, 100]$

Fig.3 Evolution curves of fitness value for F_3 , bounds $\in [-100, 100]$

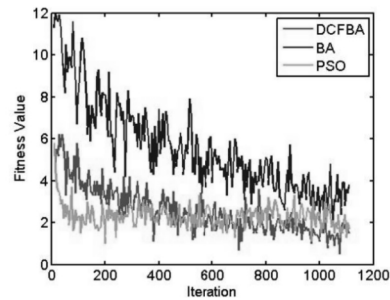


图 4 F_4 适应度值演化曲线 搜索范围为 $[-100, 100]$

Fig.4 Evolution curves of fitness value for F_4 , bounds $\in [-100, 100]$

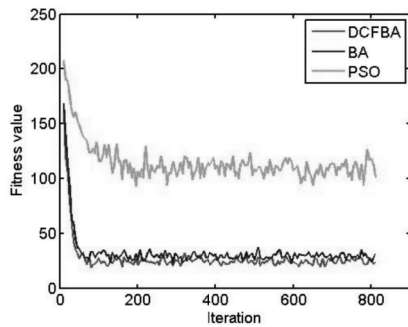


图 5 F_5 适应度值演化曲线 搜索范围为 $[-100, 100]$

Fig.5 Evolution curves of fitness value for F_5 , bounds $\in [-100, 100]$

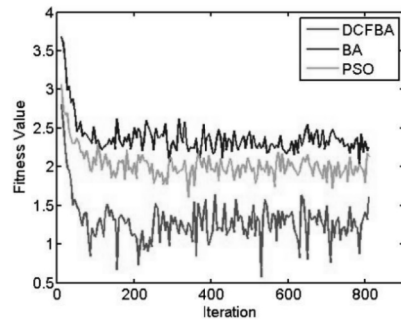


图 6 F_6 适应度值演化曲线 搜索范围为 $[-32, 32]$

Fig.6 Evolution curves of fitness value for F_6 , bounds $\in [-32, 32]$

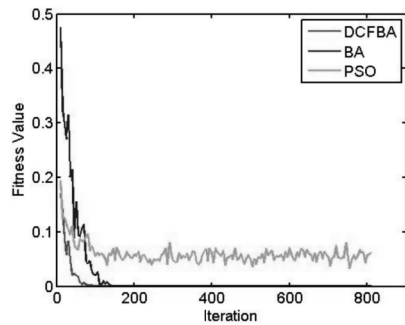


图 7 F_7 适应度值演化曲线 搜索范围为 $[-100, 100]$

Fig.7 Evolution curves of fitness value for F_7 , bounds $\in [-100, 100]$

图 8 F_8 适应度值演化曲线 搜索范围为 $[-5, 5]$

Fig.8 Evolution curves of fitness value for F_8 , bounds $\in [-5, 5]$

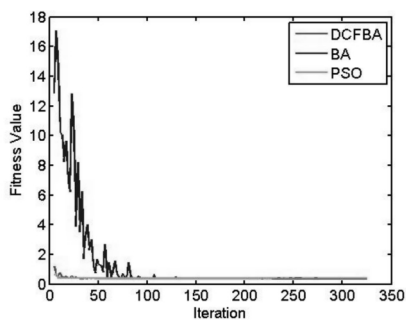


图9 F_9 适应度值演化曲线 搜索范围为 $[-5, 15]$

Fig.9 Evolution curves of fitness value for F_9 bounds $\in [-5, 15]$

5 结语

本文提出了一种新的改进蝙蝠算法——双核因素蝙蝠算法.通过 3 种算法在 9 个基准函数的仿真实验表明 DCFBA 有效可行.虽然在实验中,DCFBA 算法有比其他两种算法更好的收敛速度,但在一些迭代中是不稳定的.作为一个优秀的优化算法,DCFBA 可以尝试与其他优化算法结合,这可能会改善其优化性能.此外,DCFBA 参数设置的合理性、函数维度设置对 DCFBA 的影响等,值得进一步研究.

参 考 文 献

- [1] Bishop J M. Stochastic searching networks [C]//IEEE. 1989 First IEEE International Conference on IEEE. California: IEEE, 1989:329-331.
- [2] Dorigo M. Optimization, learning and natural algorithms [D]. Milano: Politecnico di Milano Italy, 1992.
- [3] Kennedy J, Eberhart R. Particle swarm optimization [C]//IEEE. IEEE International Conference on Neural Networks Proceedings (IEEE 4). New York: IEEE, 2002:1942-1948.
- [4] Shi Y, Eberhart R. A modified particle swarm optimizer [M]. IEEE. IEEE International Conference on Evolutionary Computation. Anchorage: IEEE, 1998: 69-73.
- [5] Yang Xinshe. A new metaheuristic bat-inspired algorithm [J]. Computer Knowledge and Technology, 2010, 284: 65-74.
- [6] Nasuto S, Bishop J M. Stabilizing swarm intelligence search via positive feedback resource allocation [M]. Berlin, Heidelberg: Springer, 2008:115-123.
- [7] Nasuto S, Bishop J M, Lauria S. Time complexity analysis of the stochastic diffusion search [M]. Berlin, Heidelberg: Springer, 1998: 260-266.
- [8] Nasuto S, Bishop J M. Convergence analysis of stochastic diffusion search [J]. Parallel Algorithms and Applications, 1999, 14(2): 89-107.
- [9] Myatt D R, Bishop J M, Nasuto S. Minimum stable convergence criteria for stochastic diffusion search [M]. Berlin, Heidelberg: Springer, 2004, 40(2): 112-113.
- [10] Parsons S. Ant Colony Optimization by Marco Dorigo and Thomas Stützle [J]. Knowledge Engineering Review, 2005, 20(1): 92-93.
- [11] Griffin D R, Webster F A, Michael C R. The echolocation of flying insects by bats [J]. Animal Behaviour, 1960, 8(3): 141-154.
- [12] Parsopoulos K E, Vrahatis M N. Recent approaches to global optimization problems through particle swarm optimization [J]. Natural Computing, 2002, 1(2-3): 235-306.
- [13] Clerc M. The swarm and the queen: Towards a deterministic and adaptive particle swarm optimization [C]// IEEE. Proceedings of the Congress on Evolutionary Computation. Piscataway, NJ: IEEE Service Center, 1999: 1951-1957.
- [14] Khan K, Nikov A, Sahai A. A fuzzy bat clustering method for ergonomic screening of office workplaces [C]//AINSC. Third International Conference on Software, Services and Semantic Technologies S3T 2011. Berlin, Heidelberg: Springer, 2011:59-66.
- [15] Yang X. Bat algorithm for multi-objective optimisation [J]. International Journal of Bio-Inspired Computation, 2011, 3(5): 267-274.
- [16] Lin J H, Chou C W, Yang C H, et al. A chaotic Lévy flight bat algorithm for parameter estimation in nonlinear dynamic biological systems [J]. ResearchGate, 2012, 2(2): 56-63.
- [17] Nakamura R Y M, Pereira L A M, Costa K A, et al. BBA: A binary bat algorithm for feature selection [J]. Graphics, Patterns and Images, 2012(1): 291-297.
- [18] Xie J, Zhou Y, Chen H. Anovel bat algorithm based on differential operator and Lévy flights trajectory [J]. Computational Intelligence and Neuroscience, 2013(2013): 38-45.
- [19] Zhang J W, Wang G G. Image matching using a bat algorithm with mutation [J]. Applied Mechanics and Materials, 2012, 203(1): 88-93.

(下转第 158 页)

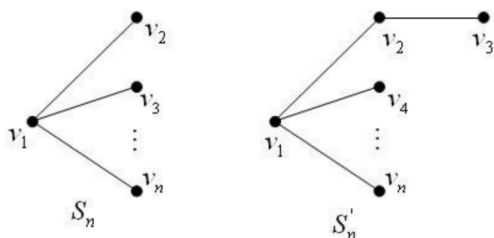


Fig.4 The graphs S_n and S'_n

图 4 图 S_n 和 S'_n

- (i) $EE(T) < EE(S'_n) < EE(S_n)$ [5].
- (ii) $SLEE(T) < SLEE(S'_n) < SLEE(S_n)$.

References

[1] Bollobás B. Modern graph theory [M]. Berlin, New York: Springer-Verlag, 1998.

[2] Estrada E. Characterization of 3D molecular structure [J]. Chemical Physics Letters, 2000, 319: 713-718.

[3] Cvetković D, Rowlinson P, Simić S K. Signless Laplacians of finite graphs [J]. Linear Algebra and Its Applications, 2007, 423: 155-171.

[4] Ayyaswamy S K, Balachandran S, Venkatakrishnan Y B, et al. Signless Laplacian Estrada index [J]. MATCH - Communications in Mathematical and in Computer Chemistry, 2011, 66: 785-794.

[5] Harary F, Palmer E M. On acyclic simplicial complexes [J]. Mathematika, 1968, 15: 115-122.

[6] Beineke L W, Pippert R E. The number of labeled k -dimensional trees [J]. Journal of Combinatorial Theory, 1969, 6(2): 200-205.

[7] Estes J, Wei B. Sharp bounds of the Zagreb indices of k -trees [J]. Journal of Combinatorial Optimization, 2014, 27: 271-291.

[8] Wang S, Wei B. Multiplicative Zagreb indices of k -trees [J]. Discrete Applied Mathematics, 2015, 180: 168-175.

[9] Wang X X, Zhai M Q, Shu J L. Upper bounds on the spectral radius of k -trees [J]. Applied Mathematics- Journal of Chinese Universities Series a, 2011, 26(2): 209-214.

[10] Huang F, Wang S. On maximum Estrada indices of k -trees [J]. Linear Algebra and Its Applications, 2015, 487: 316-327.

[11] Song L, Staton W, Wei B. Independence polynomials of k -tree related graphs [J]. Discrete Applied Mathematics, 2010, 158: 943-950.

[12] Nasiri R, Elahi H R, Fath-Tabar G H, et al. The signless Laplacian Estrada index of tricyclic graphs [EB/OL]. [2013-11-30]. <http://arxiv.org/abs/1412.2280v2>.

[13] Du Z, Liu Z. On the Estrada and Laplacian Estrada indices of graphs [J]. Linear Algebra and Its Applications, 2011, 435: 2065-2076.

[14] Ellahi H, Nasiri R, Fath-Tabar G, et al. On maximum signless Laplacian Estrada indices of graphs with given parameters [EB/OL]. [2013-11-30]. <http://arxiv.org/abs/1406.2004v1>.

(责任编辑 曹 东)

(上接第 137 页)

[20] Xie J, Zhou Y. A novel hybrid bat algorithm with harmony search for global numerical optimization [J]. Journal of Applied Mathematics, 2013(3): 233-256.

[21] Fister I, Fister D, Yang X S. A hybrid bat algorithm [J]. Electrotechnical Review, 2013, 80(1): 64-73.

[22] Fister I. Using the quaternion's representation of individuals in swarm intelligence and evolutionary computation [J]. Computer Science, 2013(1): 34-47.

[23] Li L, Zhou Y. A novel complex-valued bat algorithm [J]. Neural Computing and Applications, 2014, 25(6): 1369-1381.

[24] Metzner W. Echolocation behaviour in bats [J]. Science Progress, 1991, 75: 453-465.

[25] Yang X S, Deb S. Eagle strategy using Lévy walk and firefly algorithms for stochastic optimization [J]. Studies in Computational Intelligence, 2010, 284: 101-111.

(责任编辑 曹 东)