

DOI:10.3969/j.issn.1673-4785.201205011

网络出版地址: <http://www.cnki.net/kcms/detail/23.1538.TP.20121116.1700.002.html>

交互学习的粒子群优化算法

秦全德¹, 李丽¹, 程适^{2,3}, 李荣钧⁴

(1. 深圳大学 管理学院, 广东 深圳 518060; 2. 英利利物浦大学 电气电子工程系, 英国 利物浦 L69 3GJ; 3. 西交利物浦大学 电气电子工程系, 江苏 苏州 215123; 4. 华南理工大学 工商管理学院, 广东 广州 510640)

摘要:分析基本的粒子群优化学习机制的缺陷, 启发于人类社会不同群体之间可以交互学习的特点, 提出了一种改进粒子群优化算法——ILPSO. 在 ILPSO 算法中, 粒子由 2 个种群构成. 当 2 个种群中最佳的全局最优位置在连续一定的迭代次数内没有改善时, 执行交互学习策略. 依据每个种群的全局最优位置的适应值, 运用模拟退火的机制和轮盘赌的方法确定学习种群和被学习种群. 提出了一个基于适应度排序的经验公式, 计算学习种群中的每个粒子向被学习种群学习的概率. 为了摆脱选择压力, 采用了一种速度变异的方法. 多个测试函数的数值实验结果表明, ILPSO 具有较好的全局搜索能力, 是一种求解复杂问题的有效方法.

关键词:粒子群优化算法; 交互学习; 学习策略; 学习行为; 群体多样性

中图分类号: TP18 **文献标志码:** A **文章编号:** 1673-4785(2012)06-0547-07

Interactive learning particle swarm optimization algorithm

QIN Quande¹, LI Li¹, CHENG Shi^{2,3}, LI Rongjun⁴

(1. College of Management, Shenzhen University, Shenzhen 518060, China; 2. Department of Electrical Engineering and Electronics, Liverpool University, Liverpool L69 3GJ, UK; 3. Department of Electrical and Electronics Engineering, Xi'an Jiaotong-Liverpool University, Suzhou 215123, China; 4. School of Business Administration, South China University of Technology, Guangzhou 510640, China)

Abstract: Analyzing the drawbacks of learning mechanism in the basic particle swarm optimization (PSO), an interactive learning particle swarm optimization (ILPSO) is presented, which is inspired by the phenomenon in human society that individuals in different groups can learn each other. Particles are composed of two populations in ILPSO. When the best particle's fitness value of two populations does not improve within a certain number of successive iterations, interactive learning strategies are implemented. According to the best particle's fitness value of each population, a simulated annealing mechanism and roulette method are used to identify the learning population and the learned population. This paper proposes an empirical formula of sorting fitness value to calculate the probability of each particle in the learning population learning from the learned population. In order to escape selection pressure, a speed mutation method is used. The numerical experimental results of some benchmark functions show that ILPSO has good global search capability and is an effective method for solving complicated problems.

Keywords: particle swarm optimization algorithm; interactive learning; learning strategy; learning behavior; population diversity

粒子群优化 (particle swarm optimization, PSO) 算法是一种基于种群搜索的随机优化技术, 其模拟

了鸟群觅食过程中的迁徙和群集行为^[1]. PSO 算法具有概念简单、控制参数少、收敛速度快和易于编程实现的优点^[2], 自提出以来受到广大学者的关注. 但 PSO 算法同其他的随机搜索方法类似, 在求解复杂多峰函数时, 容易陷入局部最优^[3]. 为了提高算

收稿日期: 2012-05-07. 网络出版日期: 2012-11-16.

基金项目: 国家自然科学基金资助项目 (71071057, 71001072); 广东省自然科学基金资助项目 (S2011010001337).

通信作者: 秦全德. E-mail: qinquande@gmail.com.

法性能,较多学者提出了基于不同思想的改进算法,可以简单归纳为以下几类:1)调节算法的参数^[4-5];2)群体拓扑结构的改进^[6-7];3)与其他优化算法混合^[8-9];4)嵌入生物行为机制^[10-11];5)设计新的学习策略^[12-13].

在基本 PSO 算法中,粒子通过向个体最优位置和群体最优位置学习并不断调整其飞行速度和所在位置,从而实现在搜索空间寻优.这样的学习策略使得群体内的信息交流速度快,但由于学习方向的单一性,容易产生“趋同”现象,在算法性能上表现为迭代后期搜索缓慢甚至停滞,容易陷入局部最优^[14].因此,设计新的粒子学习策略是提高 PSO 算法性能的一个重要途径.目前,国内外学者已经在这方面开展了一些研究. Liang 等提出了广泛学习的 PSO 算法,其每个粒子根据学习概率来决定向自身个体最优位置还是其他的个体最优位置进行学习^[12].实验结果表明,采用广泛学习策略能够探索更大的搜索空间,保持了群体多样性,算法在搜索过程中不容易陷入局部最优,适合于多峰函数的求解.根据多精英比单精英更能够引导群体学习的社会现象, Huang 等提出了基于榜样学习的改进 PSO 算法^[13].文献[15]引入了超球坐标的方式来更新粒子速度,设计了一种集成学习 PSO 算法. Wang 等提出了一种自适应学习的 PSO 算法,该算法依据搜索的进程进行调整学习概率,在具有不同优点的4种学习策略中进行自适应选择^[16]. Peram 等提出了适应度值-距离-比例的 PSO 算法,此算法中每个粒子的每一维根据适应度值-距离-比例原则确定一个新的学习对象^[17].

人类社会中不同群体之间存在资源和特质的差异性,个体不但会吸收所在群体的经验,而且还会同其他群体进行信息交流.因此,人类社会的学习行为可以发生在个体与个体、个体与群体以及群体与群体之间.启发于人类社会的学习行为特点,本文提出了一种由2个种群组成的交互学习的粒子群优化算法(interactive learning particle swarm optimization, ILPSO).当2个种群中最佳的全局最优位置在连续一定的迭代次数内没有改善时,开始实施交互学习策略.交互学习的策略使得粒子学习的方向具有了多样性,为算法摆脱局部最优提供了新的额外动力.仿真实验结果表明,ILPSO 在搜索的过程中能够较好地保持群体的多样性,具有较好的全局寻优能力,是一种求解复杂问题的有效方法.

1 粒子群优化算法框架

PSO 算法中的每个粒子视为问题的一个可行

解.在 D 维的搜索空间中,粒子 i 在 t 次迭代时的状态属性由位置向量 $X_i^t = [x_{i_1}^t \ x_{i_2}^t \ \cdots \ x_{i_D}^t]$ 和速度向量 $V_i^t = [v_{i_1}^t \ v_{i_2}^t \ \cdots \ v_{i_D}^t]$ 进行描述. $x_{i_d}^t \in (l_d, u_d)$, $u_{i_d}^t \in (v_{\min}, v_{\max})$, 其中 l_d 和 u_d 分别为搜索空间的下限和上限, $d = 1, 2, \dots, D$, v_{\min} 和 v_{\max} 是由用户设定的粒子飞行的最小和最大速度.粒子的好坏由一个事先设定的适应度函数来确定.在算法的迭代过程中,每一个粒子通过向个体最优位置和群体最优位置进行学习,按照式(1)和式(2)更新飞行速度和位置,从而在搜索空间内寻优.

$$v_{i_d}^{t+1} = \omega v_{i_d}^t + c_1 r_1 (p_{i_d}^t - x_{i_d}^t) + c_2 r_2 (p_{g_d}^t - x_{i_d}^t), \quad (1)$$

$$x_{i_d}^{t+1} = x_{i_d}^t + v_{i_d}^{t+1}. \quad (2)$$

式中: $P_i^t = [p_{i_1}^t \ p_{i_2}^t \ \cdots \ p_{i_D}^t]$ 表示粒子 i 的个体最优位置; $P_g^t = [p_{g_1}^t \ p_{g_2}^t \ \cdots \ p_{g_D}^t]$ 是整个粒子群体的最优位置; r_1, r_2 为均匀分布在 $[0, 1]$ 的随机数; c_1, c_2 为正的加速系数; ω 称为惯性权重,用于控制先前速度对于当前速度的影响,从而协调粒子搜索的开发(exploration)和开采(exploitation). Shi 等经过多组实验,提出 ω 按照如式(3)线性递减变化^[4]:

$$\omega = (\omega_s - \omega_e) \times ((t_{\max} - t) / t_{\max}) + \omega_e. \quad (3)$$

式中: t_{\max} 和 t 分别表示最大迭代次数和当前的迭代次数, ω_s 和 ω_e 分别表示算法初始和停止时的惯性权重.文献[4]的研究结果表明当 $\omega_s = 0.9$, $\omega_e = 0.4$ 时,算法具有较好的性能.将文献[4]这种惯性权重按照式(3)递减的 PSO 算法称为标准粒子群优化算法(SPSO).

2 交互学习的粒子群优化算法

2.1 交互学习策略

在 ILPSO 中,粒子分成 Swarm_1 和 Swarm_2 2 个种群.算法初始运行时,2 个种群均按照式(1)更新速度. Swarm_1 中粒子 i 在 t 次迭代的位置向量、速度向量、个体最优位置和群体最优位置分别记为 $X_{i,1}^t$ 、 $V_{i,1}^t$ 、 $P_{i,1}^t$ 和 $P_{g,1}^t$.将上述4个向量的下标“1”改成“2”就代表 Swarm_2 相对应的向量.设 $P_{g,1}^t$ 、 $P_{g,2}^t$ 的适应度表示为 $fP_{g,1}^t$ 和 $fP_{g,2}^t$, P_g^t 表示 t 次迭代时2个种群中最佳的全局最优位置,其适应度表示为 $fP_g^t = \min(fP_{g,1}^t, fP_{g,2}^t)$.当 P_g^t 在连续的 k 次迭代中如果没有改善时,即 $fP_g^t - fP_g^{t-k} = 0$,表明算法陷入了局部最优,2 个种群开始实施交互学习策略,其主要步骤如下:

1)确定学习种群和被学习种群.启动交互学习策略时,其中一个种群将仍然维持原有的方式进行

寻优,称之为被学习种群;改变原有的学习方式的学习种群称为学习种群. 根据 $fP_{g,1}^t$ 和 $fP_{g,2}^t$ 的值,采用模拟退火的方法确定 Swarm_1 和 Swarm_2 作为被学习种群的概率. 这样的方式使得群体最优位置相对较差的种群将以一定的概率作为被学习种群,从而增加了算法的全局搜索能力. 用 ps 表示 Swarm_1 被选择作为被学习种群的概率,依据模拟退火算法的原理易有式(4):

$$ps = \frac{e^{-\frac{(P_{g,1} - P_g)}{T}}}{\sum_{j=1}^2 e^{-\frac{(P_{g,j} - P_g)}{T}}}. \quad (4)$$

式中: T 表示模拟退火的温度. 依据 ps 的大小,采用轮盘赌的抽样方法确定学习种群和被学习种群. 抽取在 $[0, 1]$ 均匀分布的随机数,小于 ps 时, Swarm_1 为被学习种群,反之 Swarm_2 将作为被学习种群.

2) 确定学习种群中每个粒子向被学习种群学习的概率. 在 ILPSO 中,根据学习种群中单个粒子适应度值的排序来决定学习概率的大小. 学习种群中适应度好的粒子向被学习种群学习的概率相对较小,以尽量维持目前良好的搜索形态;反之学习种群中适应度值较差的粒子,学习概率较大. 依据经验公式(5)确定学习种群中粒子 i 向被学习种群的学习概率 pc_i .

$$pc_i = 0.1 + 0.5 \times \left(\frac{\text{order}_i}{m} \right)^5. \quad (5)$$

式中: order_i 表示按粒子 i 在学习种群内适应度的排序; m 表示学习种群的规模.

3) 更新每个种群的速度和位置. 被学习种群的粒子按照式(1)更新速度;在学习种群中,如果产生的随机数小于 pc_i , 粒子 i 同时向个体最优位置、群体最优位置和被学习种群的群体最优位置学习更新速度;反之仍然根据式(1)更新速度. 当 Swarm_1 是学习种群时,向 Swarm_2 学习的粒子速度按照式(6)更新. 而当 Swarm_2 是学习种群时,向 Swarm_1 学习的粒子速度按照式(7)更新.

$$v_{i,d,1}^{t+1} = \omega v_{i,d,1}^t + c_{11} r_{11} (p_{i,d,1}^t - x_{i,d,1}^t) + c_{12} r_{12} (p_{g,d,1}^t - x_{i,d,1}^t) + c_{13} r_{13} (p_{g,d,2}^t - x_{i,d,1}^t), \quad (6)$$

$$v_{i,d,2}^{t+1} = \omega v_{i,d,2}^t + c_{11} r_{21} (p_{i,d,2}^t - x_{i,d,2}^t) + c_{12} r_{22} (p_{g,d,2}^t - x_{i,d,2}^t) + c_{13} r_{23} (p_{g,d,1}^t - x_{i,d,2}^t). \quad (7)$$

式中: r_{ij} ($i = 1, 2, j = 1, 2, 3$) 表示在 $[0, 1]$ 内均匀分布的随机数.

2.2 变异算子

如果 Swarm_1 和 Swarm_2 的群体最优位置的适应度值相差较大时,式(4)会给算法带来较大的选择压力,不利于全局搜索. 基于这样的考虑,本文采用

在任意一个种群中随机选择一个粒子在任一维上的速度按式(8)发生变异:

$$v_{i,d}^t = \begin{cases} 0.5 \times v_{\max} \times r_4, & r_3 < 0.5; \\ -0.5 \times v_{\max} \times r_4, & \text{其他.} \end{cases} \quad (8)$$

式中: v_{\max} 表示粒子飞行的最大速度, r_3 和 r_4 表示均匀分布在 $[0, 1]$ 的随机数. 由于只选择一个种群中的一个粒子一维的速度发生变异,因此几乎不会破坏群体的结构. 但每一个种群、每一个粒子和其每一维速度从统计意义上来说是以同样的概率被选择发生变异.

2.3 ILPSO 主要步骤

ILPSO 的主要步骤如下:

- 1) 置 $t = 0$; 初始化 2 个种群粒子的位置和速度,并设定相应的参数;
- 2) 计算粒子适应度值,确定个体最优位置和群体最优位置;
- 3) $fP_g^t - fP_g^{t-k} = 0$ & $t \geq k$ 是否满足,若满足执行则 4), 否则直接转到 6);
- 4) 确定学习种群和被学习种群;
- 5) 计算学习种群中每个粒子向被学习种群学习的概率;
- 6) 更新 2 个种群的粒子速度和位置;
- 7) 执行变异算子;
- 8) $t = t + 1$; 计算粒子适应度,更新每个粒子的个体最优位置和群体最优位置;
- 9) 判断程序终止条件是否满足,若满足则算法终止,输出最优解,否则转到 3).

3 仿真实验

3.1 测试函数

本文选取了 11 个常用的测试函数进行仿真实验. 其中基本测试函数包括 f_1, f_2 2 个单峰函数和 f_3, f_6, f_4 4 个多峰函数, f_7, f_8 和 f_9, f_{11} 分别是偏移测试函数和旋转测试函数. 基本测试函数的局部极值点多沿着平行的坐标轴,且全局最优位于原点,不能较好地反映现实中优化问题的复杂性. Suganthan 等对基本测试函数进行了偏移和旋转^[18]. f_7, f_8 是将基本测试函数 f_2 和 f_3 的全局最优点随机移动到每一维具有不同数值的新位置,并且加上一个偏置值 O . 本文依据文献[18]的方法进行偏移. 旋转测试函数 $f_9 \sim f_{11}$ 是通过将一个正交矩阵左乘 f_2, f_3 和 f_5 而得到的,其各个变量之间变得都不可分. 其中正交矩阵的产生采用 Salomon 在 1996 年提出的方法^[19]. 表 1 给出了每个函数的名称、数学表达式、搜索范围和最优值.

表1 测试函数及其参数设置

Table 1 Benchmark function and their settings

函数名称	函数数学表达式	搜索范围	最优值
Schwefel's Problem 1.2	$f_1(x) = \sum_{i=1}^n (\sum_{j=1}^i x_j)^2$	$(-100, 100)^D$	0
Rosenbrock	$f_2(x) = \sum_{i=1}^n (100(x_{i+1} - x_i)^2 + (x_i - 1)^2)$	$(-30, 30)^D$	0
Rastrigin	$f_3(x) = \sum_{i=1}^n (x_i^2 - 10\cos(2\pi x_i) + 10)$	$(-5.12, 5.12)^D$	0
Ackley	$f_4(x) = -20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^n x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^n \cos 2\pi x_i\right) + 20 + e$	$(-32, 32)^D$	0
Griewank	$f_5(x) = \frac{1}{4000}\sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	$(-600, 600)^D$	0
Weierstrass	$f_6(x) = \sum_{i=1}^D \sum_{k=0}^{20} [a^k \cos(2\pi b^k(x_i + 0.5))] - D \sum_{k=0}^{20} [a^k \cos(2\pi b^k \cdot 0.5)], a=0.5, b=3$	$(-0.5, 0.5)^D$	0
Shifted Rosenbrock	$f_7(x) = \sum_{i=1}^{D-1} (100(\gamma_{i+1} - x_i)^2 + (\gamma_i - 1)^2)$	$(-100, 100)^D$	390
Shifted Rastrigin	$f_8(x) = \sum_{i=1}^D (\gamma_i^2 - 10\cos(2\pi\gamma_i) + 10)$	$(-5, 5)^D$	-330
Rotated Rosenbrock	$f_9(x) = \sum_{i=1}^{D-1} (100(\gamma_{i+1} - x_i)^2 + (\gamma_i - 1)^2)$	$(-30, 30)^D$	0
Rotated Rastrigin	$f_{10}(x) = \sum_{i=1}^D (\gamma_i^2 - 10\cos(2\pi\gamma_i) + 10)$	$(-5.12, 5.12)^D$	0
Rotated Griewank	$f_{11}(x) = \frac{1}{4000}\sum_{i=1}^n \gamma_i^2 - \prod_{i=1}^D \left(\frac{\gamma_i}{\sqrt{i}}\right) + 1$	$(-600, 600)^D$	0

注:维数 D 为 30.

3.2 实验参数设置

将 ILPSO 同 SPSO、PSOPC、FDR-PSO 和 HPSO-TVAC 的性能进行比较. 为公平起见, 各种比较算法的粒子数量设置为 60, 在 ILPSO 中 Swarm_1 和 Swarm_2 的粒子数量分别设置为 30. SPSO、PSOPC、FDR-PSO、HPSO-TVAC 和 ILPSO 的详细参数根据文

献[4、10、17、20]进行设置, 具体见表2, 其中 Range 表示搜索范围的大小. 式(4)中初始温度 T_0 和退温方式根据文献[21]提出的如式(9)、(10)确定:

$$T_0 = -fp_g / \ln(0.2), \quad (9)$$

$$T_{k+1} = \lambda T_k, \quad (10)$$

式中: λ 称为退温速率, 本文取 $\lambda = 0.9$.

表2 参数设置

Table 2 Parameter settings of involved algorithms

比较算法	算法参数设置	参考文献
SPSO	$\omega: 0.9 \sim 0.4, c_1 = c_2 = 2, v_{\max} = 0.5 \times \text{Range}$	[4]
FDR-PSO	$\omega: 0.9 \sim 0.4, c_1 = c_2 = 1, c_3 = 2, v_{\max} = 0.2 \times \text{Range}$	[17]
PSOPC	$\omega: 0.9 \sim 0.7, c_1 = c_2 = 0.5, c_3: 0.4 \sim 0.6, v_{\max} = 0.2 \times \text{Range}$	[10]
HPSO-TVAC	$\omega: 0.9 \sim 0.4, c_1 = 2.5 \sim 0.5, c_2 = 0.5 \sim 2.5, v_{\max} = 0.5 \times \text{Range}$	[20]
ILPSO	$\omega: 0.9 \sim 0.4, c_1 = 2.5 \sim 0.5, c_2 = 0.5 \sim 2.5, c_{11} = c_{12} = 1, c_{13} = 2, k = 10, v_{\max} = 0.5 \times \text{Range}$	本文

3.3 实验结果与分析

将 ILPSO 和其他的 4 种 PSO 算法在 11 个测试函数上分别独立运行 25 次, 最大迭代次数为 6 000

次. 比较的 5 种 PSO 算法的实验结果(平均值和标准差)如表3, 其中最好的实验结果加粗表示. 图1描述了比较的 5 种 PSO 算法求解 $f_1 \sim f_3, f_5 \sim f_{10}$ 平均

最优值的变化曲线.

表 3 各种算法的测试结果比较
Table 3 Comparisons of experimental results

函数	指标	比较的 PSO 算法				
		SPSO	PSOPC	FDR-PSO	HPSO-TVAC	ILPSO
f_1	平均值	5.88×10^{-1}	4.37×10^{-2}	1.03×10^{-8}	1.79×10^{-7}	2.50×10^{-10}
	标准差	4.45×10^{-1}	8.37×10^{-2}	2.28×10^{-8}	2.23×10^{-7}	2.37×10^{-10}
f_2	平均值	3.07×10^1	2.01×10^1	2.19×10^1	4.22×10^{-1}	7.64×10^{-1}
	标准差	2.61×10^1	1.78×10^1	2.95×10^1	8.29×10^{-1}	6.57×10^{-1}
f_3	平均值	1.98×10^1	2.00×10^1	1.83×10^1	6.92	1.99×10^{-1}
	标准差	5.35	9.11	6.04	4.23	2.19×10^{-1}
f_4	平均值	6.22×10^{-15}	4.09×10^{-15}	7.11×10^{-15}	3.71×10^{-14}	1.19×10^{-14}
	标准差	5.41×10^{-15}	1.84×10^{-15}	2.51×10^{-15}	1.27×10^{-14}	3.01×10^{-15}
f_5	平均值	1.51×10^{-2}	4.43×10^{-3}	1.41×10^{-2}	1.03×10^{-2}	7.11×10^{-4}
	标准差	2.21×10^{-2}	5.92×10^{-3}	1.66×10^{-2}	1.02×10^{-2}	2.33×10^{-3}
f_6	平均值	1.03×10^{-2}	3.99×10^{-1}	1.71×10^{-3}	9.52×10^{-14}	0
	标准差	9.83×10^{-3}	5.42×10^{-1}	2.73×10^{-3}	8.19×10^{-14}	0
f_7	平均值	5.01×10^2	4.15×10^2	4.02×10^2	4.14×10^2	3.97×10^2
	标准差	1.15×10^2	2.73×10^1	1.11×10^1	2.71×10^1	6.41
f_8	平均值	-2.69×10^2	-3.09×10^2	-3.03×10^2	-3.21×10^2	-3.30×10^2
	标准差	1.74×10^1	5.66	1.02×10^1	4.15	3.12×10^{-1}
f_9	平均值	5.96×10^1	1.15×10^1	8.83	4.75×10^1	1.62
	标准差	6.97×10^1	1.61	3.48	9.04×10^1	9.62×10^{-1}
f_{10}	平均值	7.83×10^1	5.90×10^1	6.55×10^1	2.53×10^2	4.19×10^1
	标准差	1.95×10^1	1.18×10^1	1.82×10^1	3.70×10^1	9.28
f_{11}	平均值	1.62×10^{-2}	1.23×10^{-2}	2.15×10^{-2}	1.38×10^{-2}	4.42×10^{-3}
	标准差	1.40×10^{-2}	1.32×10^{-2}	2.84×10^{-2}	1.54×10^{-2}	6.17×10^{-1}

对于 f_1 , ILPSO 的求解结果都优于其他 4 种算法. 在对“香蕉函数” f_2 的优化中, 相较于 SPSO、PSOPC 和 FDR-PSO, ILPSO 和 HPSO-TVAC 在搜索过程中没有陷入局部最优. 虽然 ILPSO 实验结果的平均值稍差于 HPSO-TVAC, 但标准差较小, 表明 ILPSO 具有较好的鲁棒性. 对于具有大量局部最优点的复杂多峰函数 f_3 和 f_6 , ILPSO 体现了很好的优化效果, 在搜索过程中一直没有陷入局部最优, 说明了交互学习策略能够摆脱局部最优的有效性. 比较的各算法在求解函数 f_4 的性能差别不大, 都搜索到比较满意的结果. 在求解 f_5 中, SPSO、PSO-FDR 和 HPSO-TVAC 的搜索精度只能达到 10^{-2} 数量级, PSOPC 的搜索精度为 10^{-3} 数量级, 而 ILPSO 的搜索精度达到了 10^{-4} 数量级且标准差最小. 对于偏移测

试函数 f_7 和 f_8 , ILPSO 求解结果的平均值和标准差都明显好于其他算法. 将函数的坐标轴进行旋转后, 求解的难度将会提高. ILPSO 对旋转测试函数 $f_9 \sim f_{11}$ 的实验结果仍表现出了好的搜索精度和鲁棒性, 表明了其具有很好的适应性. 综上分析, ILPSO 在对基本测试函数、偏移测试函数和旋转测试函数的求解中都表现了搜索精度高和鲁棒性好的特点, 特别是对复杂的函数, 其优秀的搜索性能更加突出.

在 ILPSO 中, 由于交互学习策略的作用, 每个粒子群体的多样性得到维护, 从而提高了全局搜索能力, 不容易陷入局部最优. 根据“没有免费午餐定理”, 在维护粒子群体多样性的同时也带来了收敛速度相对不够快的成本. 从图 1 中可以看出, ILPSO 的收敛速度虽然快于 SPSO 和 HPSO-TVAC, 但在—

些测试函数的优化上比 PSOPC 和 FDR-PSO 的收敛速度稍慢.

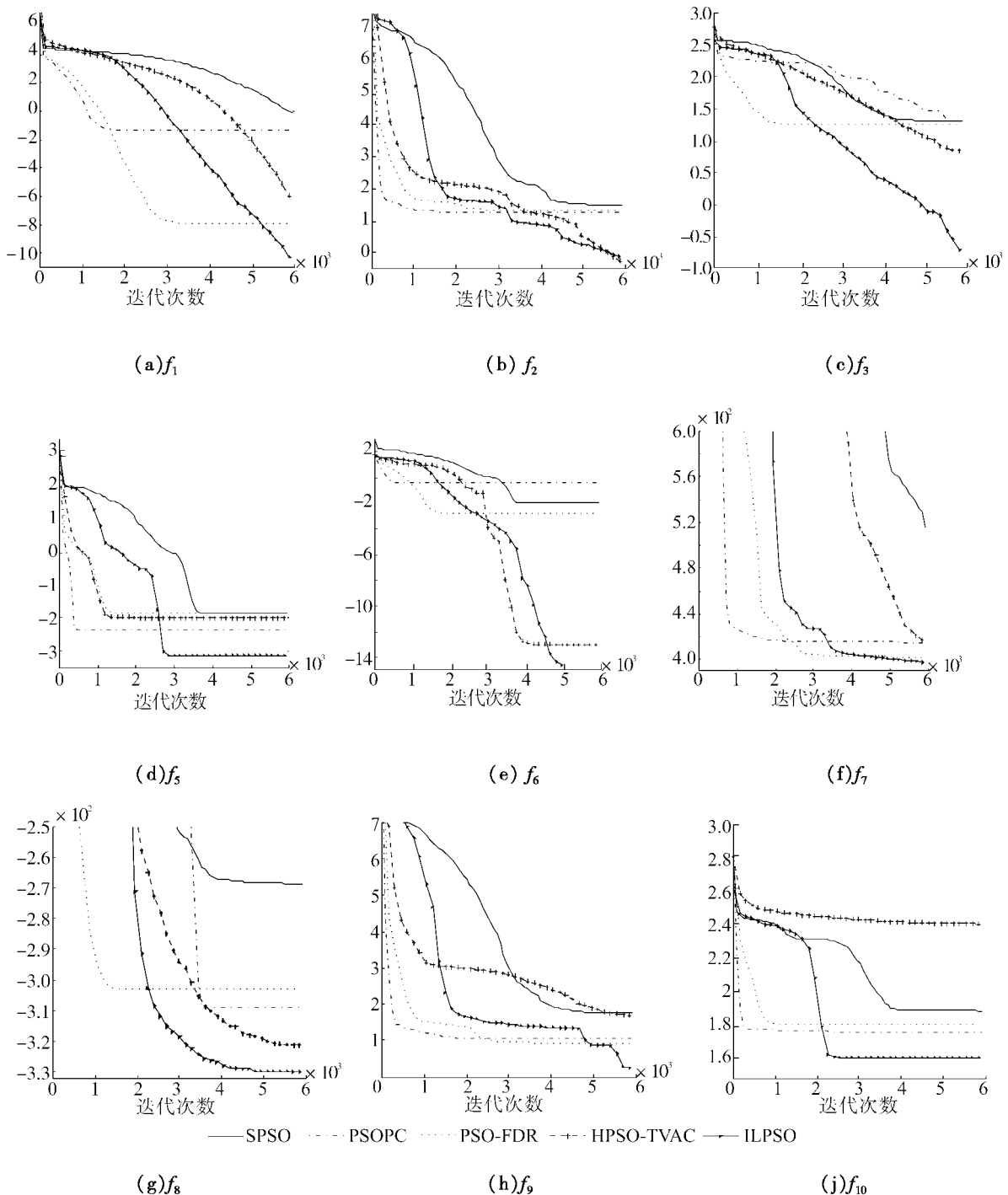


图1 各种算法实验结果的平均最优值变化

Fig.1 The convergence curve of invdvd algorithms

4 结束语

在分析基本 PSO 算法学习策略缺陷的基础上,启发于人类社会学习的特点,本文提出了一种交互学习的 PSO 算法——ILPSO. 交互学习策略中学习种群和被学习种群在搜索的过程中能够相互转换,且改善了粒子学习方向“单一性”的缺陷,保证了群体的多样性,从而提高了算法的全局搜索能力. 多个

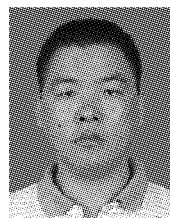
测试函数的实验结果表明 ILPSO 具有较高的搜索精度和鲁棒性,特别在求解复杂的问题中,体现的优化性能更加突出. 如何在保证 ILPSO 搜索精度和鲁棒性的同时,并进一步提高收敛速度,以及将 ILPSO 应用到实际问题的求解中是未来研究的重点.

参考文献:

[1] EBERHART R C, KENNEDY J. Particle swarm optimiza-

- tion [C]//IEEE International Conference on Neural Networks. Perth, Australia, 1995: 1942-1948.
- [2] KENNEDY J, EBERHART R C. Empirical study of particle swarm optimization [C]//Proc of Congress on Evolutionary Computation. Washington, DC, USA, 1999: 1945-1949.
- [3] ANGELINE P J. Evolutionary optimization versus particle swarm optimization and philosophy and performance difference [C]//Proc of 7th Annual Conference on Evolutionary Programming. San Diego, USA, 1998: 601-610.
- [4] SHI Y, EBERHART R C. A modified particle swarm optimizer [C]//IEEE Congress on Evolutionary Computation Anchorage. AK, NJ, 1998: 69-73.
- [5] FAN S K S, LIANG Y C, ZAHARA E. Hybrid simplex search and particle swarm optimization for the global optimization of multimodal functions [J]. Engineering Optimization, 2004, 36: 401-418.
- [6] SUGANTHAN P N. Particle swarm optimizer with neighborhood operator [C]//Proc of the IEEE Congress on Evolutionary Computation. Washington DC, USA, 1999: 1958-1961.
- [7] 杨雪榕, 梁加红, 陈凌, 等. 多邻域改进粒子群算法 [J]. 系统工程与电子技术, 2010, 32(11): 2453-2458.
YANG Xuerong, LIANG Jiahong, CHEN Ling, et al. Multi-neighbourhood improved particle swarm optimization algorithm [J]. Systems Engineering and Electronics, 2010, 32(11): 2453-2558.
- [8] 杨帆, 胡春平, 颜学峰. 基于蚁群系统的参数自适应粒子群算法及其应用 [J]. 控制理论与应用, 2010, 27(11): 1479-1488.
YANG Fan, HU Chunping, YAN Xuefeng. Particle swarm optimization algorithm of self-adaptive parameter based on ant system and its application [J]. Control Theory & Applications, 2010, 27(11): 1479-1488.
- [9] ZHANG W J, XIE X F. DEPSO: hybrid particle swarm with differential evolution operator [C]//Proc of IEEE International Conference on System, Man and Cybernetic. Washington DC, USA, 2003: 3816-3821.
- [10] HE S, WU Q H, WEN J Y, et al. A particle swarm optimizer with passive congregation [J]. BioSystems, 2004, 78: 135-147.
- [11] 秦全德, 李荣钧. 基于生物寄生行为的双种群粒子群算法 [J]. 控制与决策, 2011, 26(4): 548-552.
QIN Quande, LI Rongjun. Two-population particle swarm optimization algorithm based bio-parasitic behaviour [J]. Control and Decision, 2011, 26(4): 548-552.
- [12] LIANG J J, QIN K, SUGANTHAN P N. Comprehensive learning particle swarm optimization for global optimization of multimodal functions [J]. IEEE Transactions on Evolutionary Computation, 2006, 6(3): 281-295.
- [13] HUANG H, QIN H, HAO Z F, et al. Exemplar-based learning swarm optimization for continuous optimization [J]. Information Sciences, 2011, 182(1): 125-138.
- [14] CLERC M, KENNEDY J. The particle swarm: explosion, stability, and convergence in multi dimensional complex space [J]. IEEE Transactions on Evolutionary Computation, 2002, 6: 58-73.
- [15] SABAT S L, UDGATA S K. Integrated learning particle swarm optimizer for global optimization [J]. Applied Soft Computing, 2011, 11(1): 574-584.
- [16] WANG Y, LI B, WEISE T, et al. Self-adaptive learning based particle swarm optimization [J]. Information Sciences, 2011, 182(20): 4515-4538.
- [17] PERAM T, VEERAMACHANENI K, MOHAN C K. Fitness-distance-ratio based particle swarm optimization [C]//Proc of the Swarm Intelligence Symposium. Indianapolis, USA, 2003: 174-181.
- [18] SUGANTHAN P N, HANSEN N, LIANG J J, et al. Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization [R/OL]. <http://www.ntu.edu.sg/home/EPNSugan>.
- [19] SALOMON R. Reevaluating genetic algorithm performance under coordinate rotation of benchmark functions [J]. BioSystems, 1996, 39: 263-278.
- [20] RATNAWEERA A, HALGAMUGE S K, WATSON H C. Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients [J]. IEEE Transactions on Evolutionary Computation, 2004, 8(3): 240-255.
- [21] 王凌, 李令莱, 郑大钟. 非线性系统参数估计的一类有效搜索策略 [J]. 自动化学报, 2003, 29: 953-958.
WANG Ling, LI Lingtai, ZHENG Dazhong. A class of effective search strategies for parameter estimation of nonlinear systems [J]. Acta Automatica Sinica, 2003, 29: 953-958.

作者简介:



秦全德,男,1979年生,讲师,博士,主要研究方向为智能计算及其应用、管理决策与优化等发表学术论文10余篇。