

基于折射原理反向学习模型的 改进粒子群算法

邵 鹏^{1,2}, 吴志健^{1,2}, 周炫余², 邓长寿³

(1. 武汉大学软件工程国家重点实验室, 湖北武汉 430072; 2. 武汉大学计算机学院, 湖北武汉 430072;
3. 九江学院信息科学与技术学院, 江西九江 332005)

摘 要: 对于粒子群优化算法易陷入局部最优的缺陷, 反向学习策略对其的改进取得了较好的效果. 然而, 反向学习策略需要结合其它策略来提高算法后期的全局搜索能力, 针对此缺陷, 根据光的折射原理对反向学习策略的反向过程进行改进, 提出反向学习的统一算法模型及基于折射原理反向学习模型的改进粒子群算法. 实验与分析表明, 与其它基于反向学习的粒子群算法相比, 该模型更有效地改进了所提算法的全局搜索能力, 提高了种群的多样性, 从而提高了算法的收敛速度以及优化精度.

关键词: 智能优化算法; 粒子群优化算法; 反向学习; 折射原理

中图分类号: TP18 **文献标识码:** A **文章编号:** 0372-2112 (2015)11-2137-08

电子学报 URL: <http://www.ejournal.org.cn> **DOI:** 10.3969/j.issn.0372-2112.2015.11.001

Improved Particle Swarm Optimization Algorithm Based on Opposite Learning of Refraction

SHAO Peng^{1,2}, WU Zhi-jian^{1,2}, ZHOU Xuan-yu², DENG Chang-shou³

(1. State Key Lab of Software Engineering, Wuhan University, Wuhan, Hubei 430072, China;

2. Computer School, Wuhan University, Wuhan, Hubei 430072, China;

3. School of Information Science and Technology, Jiujiang University, Jiujiang, Jiangxi 332005, China)

Abstract: One of shortcomings found in the particle swarm optimization algorithm is that it is easy to fall into local optimum, and the opposite learning strategy has a good effect on the improvement of this shortcoming. However, to improve the global search ability by using the opposite learning strategy it is necessary that in the late algorithm other strategies are combined to opposite learning strategy. To overcome this shortcoming, this paper improves the opposite process of the opposite learning strategy according to the refraction principle of light, and proposes the unified model of opposite-based learning (UOBL) and the improved particle swarm optimization algorithm based on the opposite learning model of the principle of refraction (refrPSO). Experiment results and analysis show that the model improves the global search ability of the refrPSO algorithm more effectively compared with other particle swarm algorithm based on opposite learning and the diversity of the population. Because of these improvements, the refrPSO enhances the convergence speed and the accuracy of optimization.

Key words: intelligent optimization; particle swarm optimization; opposite-based learning; refraction principle

1 引言

粒子群优化算法^[1] (Particle Swarm Optimization, PSO) 是 Kennedy 等根据鸟类群体觅食的社会行为提出的智能优化算法, 其具有参数少、易实现等优点, 并在许多科

学工程领域有着广泛的应用^[2]. 然而, PSO 算法在优化时容易陷入局部最优及收敛精度低等不足^[3]. 对于此, 近年来学者们提出了各种不同类型的改进算法. 其中, 反向学习策略 (Opposite-Based Learning, OBL) 在 PSO 算法中的应用取得了不错的效果. Tizhoosh 和 Rahnamayan 等

分别将 OBL 应用于遗传算法^[4] (Genetic Algorithm, GA)、差分演化算法^[5] (Differential Evolution, DE) 并取得较好的优化效果. Wang 等在 PSO 算法中运用 OBL 并结合柯西变异取得了较好的效果^[6,7]. 喻飞等根据透镜成像原理提出透镜成像反向学习策略并将其应用于 PSO 算法并取得较好的效果^[8].

为了进一步提高 PSO 算法的优化性能, 本文根据光的折射原理对 OBL 的反向过程进行改进, 推导并建立几种 OBL 策略的统一模型, 并将其运用到 PSO 算法中, 提出基于折射原理反向学习模型的改进 PSO 算法. 实验结果表明, 该算法提高了种群多样性和全局搜索能力, 从而具有很快的收敛速度和很高的收敛精度且不需要结合其它策略.

2 粒子群优化算法与反向学习

2.1 粒子群优化算法 PSO

在 PSO 算法中, 每个粒子代表搜索空间中一个可能解, 粒子通过自身的经验值(个体极值 $pbest$) 和其它粒子的经验值(全局最优值 $gbest$) 来调整自身的搜索策略. 在一个 D 维的搜索空间中, 由 n 个粒子构成一个搜索群体, 第 i 个粒子的位置可以表示为 $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$, $i = 1, 2, \dots, n$; 每次迭代中粒子 i 飞行方向可以表示为 $V_i = (v_{i1}, v_{i2}, \dots, v_{iD})$, $i = 1, 2, \dots, n$. 每个粒子通过跟踪上述的 $pbest$ 和 $gbest$ 来更新速度和位置. PSO 算法的速度和位置更新方程如下:

$$v_{id}(t+1) = \omega v_{id}(t) + c_1 r_1 (pbest_{id} - x_{id}(t)) + c_2 r_2 (gbest_{id} - x_{id}(t)) \quad (1)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (2)$$

其中, $v_{id}(t+1)$ 为粒子 i 在第 $t+1$ 代的速度; $x_{id}(t+1)$ 为粒子 i 在第 $t+1$ 代的位置; c_1 和 c_2 为加速因子; ω 为惯性权重; r_1 和 r_2 为 $[0, 1]$ 间服从均匀分布的随机数.

2.2 反向学习策略

Tizhoosh 在文献[4]中给出 OBL 的定义如下:

定义 1 若 x 是定义在实数集 R 上的一个区间 $[a, b]$ 的一个实数, 即 $x \in [a, b]$, 则 x 的反向解 x_1 定义如下:

$$x_1 = a + b - x \quad (3)$$

定义 2 若 $P(x_1, x_2, \dots, x_n)$ 是 n 维空间上的一个点, x_1, x_2, \dots, x_n 为实数, 且 $x_i \in [a_i, b_i]$, 则其反向解 x_i^* 定义为

$$x_i^* = a_i + b_i - x_i, \quad i = 1, 2, 3, \dots, n \quad (4)$$

定义 2 为定义 1 在 n 维空间上的扩展. 结合上述两个定义, 其基本思想可描述为: 假设 $f(x)$ 是原始函数, $g(\cdot)$ 是评价函数. 如果 x 是区间 $[a, b]$ 产生的随机数, x_1 是 x 的反向解, 那么在每次迭代中计算 $f(x)$ 和 $f(x_1)$,

若 $g(f(x)) \geq g(f(x_1))$, 仍然以 x 进行迭代; 否则以 x_1 进行迭代. 一维空间反向学习过程如图 1 所示, 图中 O_k 为第 k 次反向的基准点, 其值为 a_i 和 b_i 的中点.

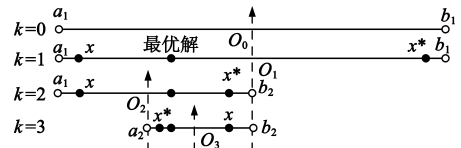


图1 一维空间反向学习过程

3 基于折射原理的反向学习模型的改进 PSO 算法

3.1 折射原理

光的折射是自然界中普遍存在的物理现象, 如图 2 所示. 其中, O 为两种介质交界点.

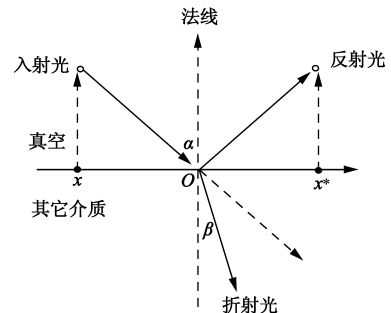


图2 光的折射现象

定义 3 光从一种介质斜射入另一种介质时, 传播方向发生变化, 这种现象称为光的折射.

定律 1 光从真空中斜射入其它介质时, 折射光线与入射光线、法线在同一平面上, 折射光和入射光分居法线两侧, 折射角小于入射角; 入射角增大(减小)时, 折射角也随之增大(减小), 称为光的折射定律.

定义 4 光从真空中射入其它介质发生折射时, 入射角 α 与折射角 β 符合斯涅尔定律(Snell's Law)^[9], 即入射角 α 与折射角 β 的正弦之比 n 称为介质的绝对折射率, 简称折射率, 用公式表示为:

$$n = \sin\alpha / \sin\beta \quad (5)$$

式(5)又称为斯涅尔公式. 若已知入射角 α , 折射率 n 固定不变, 则折射角可以表示为:

$$\beta = \arcsin(\sin\alpha / n) \quad (6)$$

式(6)中, 根据定律 1 可知当入射角 α 减小时, 折射角 β 随之减小.

3.2 基于折射原理反向学习统一模型

OBL 策略的主要思想是通过求解当前解的反向解, 并从中选取较优解替代当前解参与下一代优化. 然而, 文献[4, 8]指出, OBL 策略在算法演化前期效果较好, 但随着演化过程不断进行, 可能会出现反向解落入局部

最优值附近,而一旦有粒子落入局部最优区域,其它粒子会快速向该区域靠近,导致陷入局部最优,算法收敛精度降低,其优化效果不明显.而文献[4,6,7]中引入其它各种策略比如以某一概率下使用 OBL 策略,增加了算法的随机扰动性,但这会导致算法收敛速度变慢.为了解决 OBL 策略存在的问题,引入折射原理对 OBL 策略的反向过程进行改进.如图 3 所示,其为基于折射原理的反向学习过程示意图.下面根据图 2 和图 3 来推导基于折射原理的反向学习模型.

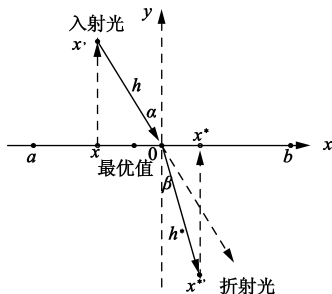


图3 基于折射原理的反向学习过程示意图

在粒子所在的一维空间中,以 x 轴为界将空间划分为上下两个部分.空间的上部分看作是自然界中的真空,下部分看作是其它介质空间.假设 x 轴上粒子的搜索区间为 $[a, b]$,即 $x \in [a, b]$, y 为法线.为了将折射原理应用于反向学习过程,作如下假设:在粒子的正上方某高度处有一光源 x' (称为入射点),其向交界点 O 发出一束入射光 $x'O$,令 $x'O$ 的距离为 h .由定义 3 和定律 1 知,光束 $x'O$ 在交界点 O 处发生折射现象,折射光束为 Ox'' ,其中 x'' 为折射点,令 Ox'' 的距离为 h^* .

定义 5 设折射点 x'' 在 x 轴上的投影为 x^* ,称 x^* 为粒子 x 基于折射原理的反向解.

折射原理反向学习中的交界点取值与反向学习一样,也为搜索区间 $[a, b]$ 的中点 $(a+b)/2$.由图 3 所示不难得出 $\sin\alpha$ 和 $\sin\beta$ 的计算表达式:

$$\sin\alpha = ((a+b)/2 - x)/h \quad (7)$$

$$\sin\beta = (x^* - (a+b)/2)/h^* \quad (8)$$

由上两式可得

$$\sin\alpha/\sin\beta = (((a+b)/2 - x)/(x^* - (a+b)/2))(h^*/h) \quad (9)$$

根据式(5)并设 $k = h/h^*$,则式(9)可改写为

$$kn = ((a+b)/2 - x)/(x^* - (a+b)/2) \quad (10)$$

由式(10)可得

$$x^* = (a+b)/2 + (a+b)/(2kn) - x/kn \quad (11)$$

式(11)即为基于折射原理的反向学习反向解的计算公式.若 $n=1$,则式(11)可以简化为:

$$x^* = (a+b)/2 + (a+b)/2k - x/k \quad (12)$$

式(12)为文献[8]中透镜成像反向学习反向解的求解公式.若 $n=1$ 且 $k=1$,则式(11)可简化为:

$$x^* = a + b - x \quad (13)$$

式(13)为文献[4]中提出的反向学习反向解的求解公式.

因此,文中将式(11)作为上述几种 OBL 策略的统一模型(UOBL).实际上,从图 2 可知,入射光在交界点不光发生折射,同时有部分光发生了反射.若入射光与反射光的长度相等,可知反射光在 x 轴上的投影点正好与图 1 中 OBL 策略所取得的反向解重合.因此,可将文献[4]中的 OBL 原理用光的反射现象来解释,从而将其归结到 UOBL 模型中.

3.3 UOBL 在 PSO 算法中的应用

为了进一步解决 PSO 算法容易陷入局部最优的不足,将上述 UOBL 模型应用在 PSO 算法中以改进其优化性能.由于 PSO 算法中粒子是多维的,因此式(11)推广到 D 维空间可改写为:

$$x_{i,j}^* = (a_j + b_j)/2 + (a_j + b_j)/(2kn) - x_{i,j}/kn \quad (14)$$

式(14)中, $x_{i,j}$ 表示算法中第 i 个粒子的第 j 维的值, $x_{i,j}^*$ 是 $x_{i,j}$ 基于折射原理的反向学习的反向解; a_j, b_j 分别是当前种群中第 j 维的最小值和最大值,这种动态的边界更新有利于搜索到更好的解,从而加快收敛速度^[10].

为了进一步提高 PSO 算法的性能,再来看看式(14)中两个参数 k (伸缩因子)和 n (微调因子,实际指 n 中的入射角 α).由图 3 及前文所述知,当调节 k 的值时,由式(14)所计算出的反向解在 x 轴的位置也发生变化.因此,可通过调节 k 的值增加算法跳出局部最优的概率. k 的取值采用线性递减的方式:

$$k = k_{\max} - (k_{\max} - k_{\min})t/t_{\max}, k \in [k_{\min}, k_{\max}] \quad (15)$$

式(15)中, k_{\max}, k_{\min} 分别为 k 的最大值和最小值, t 为当前迭代次数, t_{\max} 为最大迭代次数.

从图 3 知,若 n 固定,当入射角减小时,入射光在 x 轴上的投影点也随之变化.若入射角减小时的投影点对应的适应值优于其先未减小时的适应值,则将该投影点代替入射角减小前的点加入种群进行迭代,否则入射角继续减小.这里入射角减小的目的是当伸缩因子仍不能使算法跳出局部最优时(判断准则是若在一一定的迭代次数后适应值仍未发生变化)对种群多样性进行微调,从而扩大搜索区域的范围,提高种群的多样性,加快寻优速度,提高算法的全局搜索能力.因此,入射角按式(16)进行微调:

$$\alpha(t+1) = \alpha(t) - Ft \quad (16)$$

式(16)中, $\alpha(t)$ 表示第 t 代时入射角度数, Ft 为微调值.为了进一步说明 refrPSO 算法的性能,下面运用二维空间示意图来分析 UOBL 模型改进算法性能的执行过程.

假设当前解与其反向解及最优解所在位置如图 4

所示.从图 4 可知,当前解利用 OBL 模型产生的反向解 4 离全局最优区较远,说明其可能陷入局部最优区,而利用 UOBL 模型所产生的反向解 1 同样离全局最优区较远,说明其也可能陷入局部最优区,这时通过调节 k 值使其减小所产生的反向解 2 离全局最优区近些,从而说明通过 k 值增加了跳出极值区的概率,但相对来讲仍然离全局最优区较远,或者说陷入了另一个局部最优区.此时,通过调节 n 使入射角减小导致折射角减小所产生的反向解 3 离最优解更近了,落入了全局最优区.也就是说,该模型中引入的两个参数相互配合,起到引导粒子向最优解附近靠近的作用.因此,UOBL 模型具有探索反向学习模型未搜索区域的作用,也就是说拓宽了粒子的搜索空间,提高了种群的多样性,增加了跳出局部最优区的概率.基于 UOBL 模型的改进 PSO 算法描述如下:

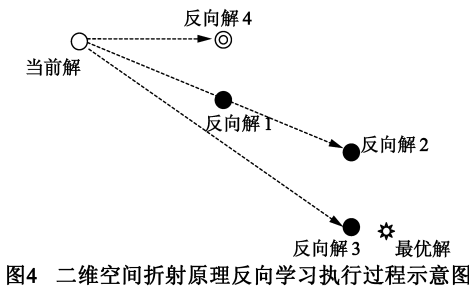


图4 二维空间折射原理反向学习执行过程示意图

算法 1 运用 UOBL 模型改进 PSO 算法 (refrPSO)

输入:相关参数等的输入;输出:最优解,收敛精度;

- (1) 随机初始化种群 P , 种群大小 $popsize$;
- (2) for $i = 1$ to $popsize$
- (3) 根据式(14)计算初始化种群 P 中粒子 x_i 的折射解 x_i^* ;
- (4) 分别计算 x_i 和 x_i^* 的适应值 $f(x_i)$ 和 $f(x_i^*)$;
- (5) if $(f(x_i) < f(x_i^*))$
- (6) 将折射解 x_i^* 代替 x_i 加入种群参与迭代;
- (7) end if
- (8) end for
- (9) while(未达到最大迭代次数或收敛精度)
- (10) 根据式(15)更新伸缩因子 k ;
- (11) 根据式(16)更新微调因子 α ;
- (12) 更新当前粒子的边界 a 和 b ;
- (13) for $i = 1$ to $popsize$
- (14) 根据式(1)更新粒子 x_i 的速度;
- (15) 根据式(14)计算初始化种群 P 中粒子 x_i 的折射解 x_i^* ;
- (16) 根据式(2)更新粒子 x_i 的位置;
- (17) if $(f(x_i) < f(x_i^*))$
- (18) 将折射解 x_i^* 代替 x_i 加入种群 P 参与迭代;
- (19) end if
- (20) end for
- (21) 更新 $gbest$ 和 $pbest$, 迭代次数加 1;
- (22) end while
- (23) 输出当前最优解.

4 实验结果与分析

4.1 参数设置

为了测试 refrPSO 算法的性能,选取文献[8,13]中的 8 个函数以及文献[6]中的第 2 个和第 3 个函数(这两个函数文献[8]中没有)共 10 个函数作为算法的测试函数.按照上述顺序编为 $F_1 \sim F_{10}$,除 F_8 最优值为 -12569.5 外,其余的最优值均为 0.另外, $F_1 \sim F_3, F_9, F_{10}$ 为单峰函数, $F_4 \sim F_8$ 为多峰函数.选取标准 PSO 以及 OPSPSO^[6]、GOPSPSO^[7]、lensPSO^[8] 等三个基于 OBL 的 PSO 算法作为对比.上述五个算法中相同参数设置如下: $\omega = 0.72984, c_1 = c_2 = 1.49$,最大迭代次数为 100000,种群大小为 10,维数为 30.refrPSO 算法中 $k = 0.75, n = 2$,入射角最大值为 $\pi/8$,微调值 Ft 的初始取值为 0.001.其它各个算法中所特有的参数按各自文献中的取值.

4.2 实验结果与分析

4.2.1 各算法精度结果与分析

表 1 中给出 5 种改进的 PSO 算法分别在 10 个测试函数上运行 30 次的结果.从表 1 中可以看出,对于 $F_1, F_2, F_4, F_5, F_6, F_9, F_{10}$ 这 7 个测试函数,refrPSO 算法精度明显优于其它四种改进 PSO 算法,特别是 F_1, F_2, F_4, F_5, F_9 这 5 个函数都找到了最优值.对于 F_3 ,由于其函数特性,极易陷入局部最优^[11,12],除 refrPSO 算法外其它算法均陷入了局部最优,而 refrPSO 算法优化的最好值却达到 0,说明算法在某些次运行时跳出局部最优并找到最优值,即整体上 refrPSO 算法精度高于其它算法.对于 F_7 ,五种算法均陷入局部最优,性能相差不大.对于 F_8 ,它是一典型的具有欺骗性的函数,其全局极值点周围有许多局部极值,算法很容易就陷入局部极值^[14].前四种算法陷入局部最优,而 refrPSO 算法却跳出了局部最优并且找到了全局最优值(部分运行次最优).因此,refrPSO 算法精度整体上高于其它算法精度.

4.2.2 平均迭代次数与运行成功率

为了进一步说明 refrPSO 算法的优化性能,文中引入平均迭代次数(NFC)和运行成功率(SR)两个评价指标.NFC 是指 30 次运行中每次运行达到所规定的算法精度的最小迭代次数之各的平均值.SR 是指 30 次运行中达到所规定的算法精度的次数与总次数之比.平均迭代次数越少、成功率越高算法的性能越好.从表 2 中可以看出,除了 F_6, F_7 和 F_{10} 外,对于其它函数 refrPSO 算法的 NFC 均远小于其它算法且 SR 远高于其它算法(F_9 的 SR 一样但 NFC 远高于其它算法).对于 F_6, F_7 和 F_{10} ,虽然五种算法的 NFC 和 SR 一样,但结合表 1 可知 refrPSO 算法的性能仍然是优于其它算法的.表 2 中 $F_1 \sim F_{10}$ 规定的精度分别为: $0, 10^{-200}, 10^{-2}, 10^{-25}, 0, 10^{-25},$

10^{-25} 、 -12569.4 、 10^{-200} 、 10^{-25} . 若 NFC 达到 100000 代, 则表示未达到规定精度.

表 1 5 种改进 PSO 的实验对比结果

函数	算法	最优值	均值	方差	最差值	平均时间(ms)
F_1	PSO	5.85e-015	7.20e-006	3.38e-005	1.85e-004	236.6
	OPSO	2.87e-044	9.59e-035	4.38e-034	2.36e-033	767.3
	GOPSO	8.79e-162	1.75e-132	9.59e-132	5.25e-131	779.5
	lensPSO	3.22e-273	1.77e-259	9.99e-260	4.56e-258	7323
	refrPSO	0	0	0	0	685.6
F_2	PSO	4.63e-002	7.34e-002	4.01e-001	2.20e+000	241.9
	OPSO	2.75e-003	2.18e+000	8.54e+000	4.58e+001	647.5
	GOPSO	2.97e-024	4.12e-003	1.71e-002	9.20e-002	772.2
	lensPSO	6.44e+001	1.12e+002	2.84e+001	1.65e+002	8267
	refrPSO	0	0	0	0	780.8
F_3	PSO	7.02e+000	2.61e+001	1.22e+001	7.78e+001	223.6
	OPSO	6.37e+000	2.31e+001	1.08e+001	7.62e+001	599.8
	GOPSO	1.97e+001	2.07e+001	4.22e-001	2.18e+001	695.2
	lensPSO	9.22e-004	9.16e+000	1.36e+001	7.76e+001	7120
	refrPSO	0	8.46e-001	4.63e+000	2.68+001	372.3
F_4	PSO	3.08e+001	5.82e+001	1.25e+001	7.95e+001	326.4
	OPSO	2.58e+001	7.45e+001	5.65e+001	2.38e+002	677.2
	GOPSO	0	1.38e+001	5.14e+001	5.26e+002	677.8
	lensPSO	1.84e+002	2.08e+002	1.41e+001	2.31e+002	9959
	refrPSO	0	0	0	0	444.8
F_5	PSO	2.16e-011	5.98e-001	8.84e-001	4.02e+000	329.4
	OPSO	0	1.57e-002	1.83e-001	6.12e-002	708.2
	GOPSO	0	0	0	0	710.6
	lensPSO	0	4.94e-002	1.56e-001	8.67e-001	11321
	refrPSO	0	0	0	0	560.9
F_6	PSO	3.46e+000	7.68e+000	2.35e+000	1.21e+001	293.4
	OPSO	9.62e-013	2.08e+000	1.05e+000	5.44e+000	680.8
	GOPSO	9.58e-013	9.58e-013	1.45e-015	9.58e-013	700.1
	lensPSO	9.58e-013	9.61e-013	1.77e-015	9.63e-013	9953
	refrPSO	4.44e-016	4.44e-016	0	4.44e-016	475.3
F_7	PSO	1.02e-001	1.49e-001	2.27e-002	2.03e-001	456.3
	OPSO	2.45e-003	2.45e-003	2.98e-011	2.45e-003	625.6
	GOPSO	2.45e-003	2.45e-003	2.78e-011	2.45e-003	633.1
	lensPSO	2.45e-003	2.45e-003	6.83e-011	2.45e-003	7485
	refrPSO	2.45e-003	2.45e-003	3.23e-017	2.45e-003	357
F_8	PSO	-7614.57	-6819.17	548.65	-5323.52	310.9
	OPSO	-11622	-11156	335.892	-10164.3	784.2
	GOPSO	-11621.5	-11018.3	385.244	-10227.9	772.9
	lensPSO	-5750.98	-4934.23	328.165	-4497.14	11310
	refrPSO	-12569.5	-9667.72	2702.34	-6055.3	562.1
F_9	PSO	1.21e-014	5.28e-005	2.31e-004	1.25e-003	239.1
	OPSO	5.06e-047	6.44e-036	3.42e-035	1.87e-034	608.8
	GOPSO	2.91e-160	1.14e-040	5.80e-040	3.17e-039	609.3
	lensPSO	9.15e-281	9.86e-161	5.40e-160	2.95e-159	7218
	refrPSO	0	3.67e-284	0	1.01e-282	647.4
F_{10}	PSO	1.64e-002	8.71e-002	9.18e-002	4.37e-001	231.3
	OPSO	1.34e-002	3.22e-002	1.19e-002	5.80e-002	609.5
	GOPSO	3.77e-003	2.26e-002	1.13e-002	5.36e-002	631.5
	lensPSO	4.13e-003	1.43e-002	7.31e-003	3.39e-002	7772
	refrPSO	1.11e-015	5.58e-006	1.21e-005	6.33e-005	350.5

表 2 各算法的 NFC 及 SR

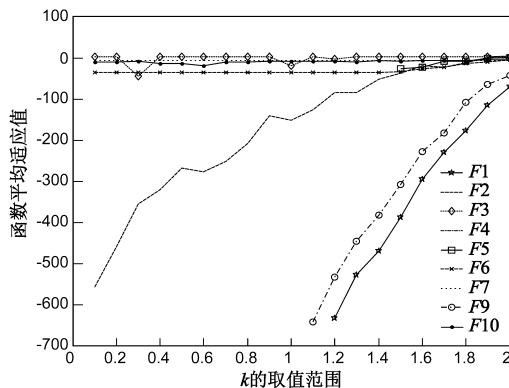
算法	指标	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}
PSO	NFC	100000	100000	100000	100000	100000	100000	100000	100000	100000	100000
	SR	0	0	0	0	0	0	0	0	0	0
OPSO	NFC	100000	100000	100000	100000	20268	100000	100000	100000	100000	100000
	SR	0	0	0	0	0.4	0	0	0	0	0
GOPSO	NFC	100000	100000	100000	54462	17307	100000	100000	100000	100000	100000
	SR	0	0	0	0.86	1	0	0	0	0	0
lensPSO	NFC	100000	100000	96800	100000	66647	100000	100000	100000	74882	100000
	SR	0	0	0.1	0	0.4	0	0	0	1	0
refrPSO	NFC	34201	9699	16862	3739	393	100000	100000	66994	3157	100000
	SR	1	1	0.87	1	1	0	0	0.43	1	0

4.2.3 相关参数对算法的影响

refrPSO 算法主要涉及两个参数: k 和 n , 现分别对其取值及对算法的影响进行分析.

(1) k 的确定及影响

为了确定 k 的值, 作如下实验: 将 k 的取值范围预估在 $[0, 2]$ 对 10 个测试函数独立运行 20 次取平均值, 实验结果如图 5 所示. 从图 5 可以看出, 大部分函数随着 k 值的逐渐减小其曲线呈下降趋势或呈下降后上升又下降的趋势. 因此, k 值采用线性递减的方式且初始值取较好区间 $[0, 1.4]$ 的近中间值 0.75. 然而, 从图 5 中也可以看出, 少部分函数在后期陷入了局部最优, 此时 k 值已经无法让其跳出局部最优了. 为了解决该问题, refrPSO 算法引入另一重要参数 n .

图 5 k 取值变化图

(2) n 的确定及影响

由于折射率 n 主要与入射角有关, 而入射角与起关键作用的微调值 F_t 有关. 前文已述, 入射角在 k 变化下仍然不能跳出局部最优时让其改变, 从而使折射点在 x 轴上的投影点即粒子位置发生改变, 这可能是较优的位置也可能是较差位置, 也就是使种群中个体多样性增加了. 微调值 F_t 减小过大, 可能跳过较优位置, 过小可能起不到跳出局部最优的作用. 为了找到合适的 F_t 值, 当 k 为 0.75 时以函数 F_8 为例作如下实验.

从表 3 可以看出, F_t 取 0.001 时取值最优, 过大或过小都会导致性能下降. 因此, 微调值 F_t 取 0.001.

表 3 F_t 不同的取值及对应的函数值

F_t 取值	函数 F_8 最优值
0.1	-5966.45
0.01	-7506.18
0.001	-12569.5
0.0001	-6677.11

4.3 算法收敛与时间复杂度分析

4.3.1 算法收敛性分析

受篇幅限制, 图 6 中只显示了 5 种改进 PSO 算法分别在 6 个测试函数上运行的收敛性情况. 从图 6 知, refrPSO 算法的收敛速度均远快于另外 4 个算法.

王晖在文献[15]中给出了基于一般反向学习的群体搜索算法收敛性的证明, 这里引用其结论来对 refrPSO 算法收敛性作简要证明. 由于 PSO 算法也是群体搜索算法的一种, 故得出以下定理.

定理 1 若基于一般反向学习的 PSO 算法收敛, 则 refrPSO 算法也是收敛的.

证明 设 $x_i(t)$ 和 $x_i^*(t)$ 分别为第 t 代当前解和其反向解, $x_{i,j}(t)$ 和 $x_{i,j}^*(t)$ 分别为 $x_i(t)$ 和 $x_i^*(t)$ 第 j 维上的值, 问题的全局最优解为 x_{best} . 由定理 1 条件知基于一般反向学习的 PSO 算法收敛, 对于第 t 代种群中的解 $x_{i,j}(t)$ 有

$$\lim_{t \rightarrow \infty} x_{i,j}(t) = x_{best,j} \quad (17)$$

由于 $a_j(t) = \min(x_{i,j}(t))$ 且 $b_j(t) = \max(x_{i,j}(t))$, 所以有

$$\lim_{t \rightarrow \infty} a_j(t) = \lim_{t \rightarrow \infty} b_j(t) = x_{best,j} \quad (18)$$

根据式(14)折射原理反向学习产生的当前解的反向解为

$$x_{i,j}^*(t) = (a_j(t) + b_j(t))/2 + (a_j(t) + b_j(t))/(2kn) - x_{i,j}(t)/kn \quad (19)$$

当上式中的代数 $t \rightarrow \infty$ 时有

$$\begin{aligned} \lim_{t \rightarrow \infty} x_{i,j}^*(t) &= \lim_{t \rightarrow \infty} (a_j(t) + b_j(t))/2 + \\ &\quad \lim_{t \rightarrow \infty} (a_j(t) + b_j(t))/(2kn) - \\ &\quad \lim_{t \rightarrow \infty} x_{i,j}(t)/kn \\ &= x_{best,j} + x_{best,j}/kn - x_{best,j}/kn = x_{best,j} \end{aligned} \quad (20)$$

由式(20)可知,当 $x_i(t)$ 收敛于 x_{best} 时,基于折射原理反向学习产生的反向解 $x_i^*(t)$ 也收敛于 x_{best} . 因此,基于一般反向学习的 PSO 算法收敛,则 refrPSO 算法也是收敛的.

4.3.2 算法时间复杂度分析

PSO 算法除了其本身的算法策略外并没有引入其它的策略,而其它四种算法引入了不同反向学习策略. 由此可知 PSO 算法运行时间最少. 对于 refrPSO 算法,其主要耗时在反向学习边界的更新、当前解与其反向解的比较上. 由算法思想知只是当前解与其反向解比较,即比较次数与种群大小相同. 对于 OPSSO 和 GOPSO 算法,主要耗时在反向学习边界的更新、最优种群的选取上; lensPSO 算法主要耗时在反向学习边界的更新、最优

种群的选取以及种群坡度的比较计算上. 若设种群大小为 N , 维数为 D , 迭代次数为 F , 根据五种算法的思想不难得出 PSO 算法比较次数为 ND , 后四种算法的反向学习边界更新均为 ND ; 最优种群选取 OPSSO 和 GOPSO 算法为 $2N^2$, lensPSO 算法为 $3N^2$; 坡度计算为 $F-1$; 原始解与其反向解为 N . 由此可知记 PSO 算法时间复杂度为 $O(ND)$, OPSSO 和 GOPSO 算法为 $O(2ND + 2N^2)$, lensPSO 算法为 $O(2ND + 3N^2 + F)$, refrPSO 算法为 $O(2ND + N)$.

综上所述, refrPSO 算法的时间复杂度优于 OPSSO、GOPSO 和 lensPSO 算法, 次于 PSO 算法, 这与表 1 中最后一列列出的各种算法 30 次运行的平均时间吻合(由于运行环境影响存在些许误差).

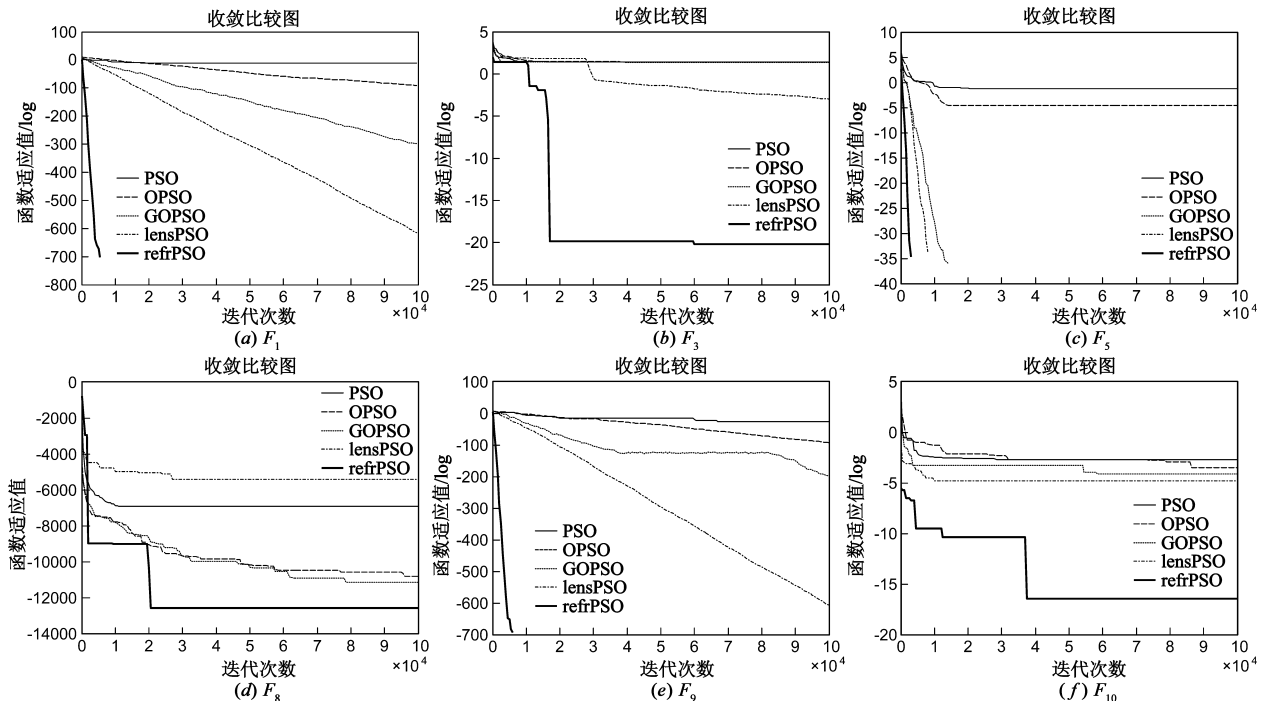


图6 5种算法的收敛曲线对比图

5 结论

将光在不同介质中传输会发生折射现象的原理应用到反向学习策略过程的改进中, 提出几种反向学习策略的统一模型(UOBL), 并将该模型应用于 PSO 算法的改进, 提出了基于折射原理的反向学习模型的改进 PSO 算法(refrPSO 算法). 通过对 refrPSO 算法的理论分析及大量的实验结果表明, UOBL 模型在 refrPSO 算法中的应用扩大了搜索区域, 提高了种群的多样性, 增加了跳出局部最优的概率, 增强了该算法的全局搜索能力, 从而不需要结合其它策略就能使 refrPSO 算法不管是对于单峰还是多峰问题都具有很快的收敛速度和很高的优化精度.

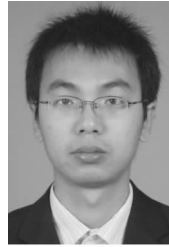
参考文献

- [1] J. Kennedy, R C Eberhart. Particle swarm optimization[A]. Proceedings of the IEEE International Joint Conference on Neural Networks[C]. IEEE, 1995. 1942 - 1948.
- [2] 胡旺, 李志蜀. 一种更简化而高效的粒子群优化算法[J]. 软件学报, 2007, 18(4): 861 - 868.
Hu Wang, Li Zhi-shu. A simpler and more effective particle swarm optimization algorithm[J]. Journal of Software, 2007, 18(4): 861 - 868. (in Chinese)
- [3] Gandomi A H, Yun G J, Yang X S, et al. Chaos-enhanced accelerated particle swarm optimization[J]. Communications in Nonlinear Science and Numerical Simulation, 2013, 18(2): 327

- 340.

- [4] Tizhoosh H. Opposition-based learning: A new scheme for machine intelligence [A]. Proceedings of International Conference on Intelligent Agent, Web Technologies and Internet Commerce [C]. Vienna: IEEE, 2005. 695 - 701.
- [5] S Rahnamayan, H R Tizhoosh, M M A Salama. Opposition-based differential evolution algorithm [A]. IEEE Congress on Evolutionary Computation [C]. Canada: IEEE, 2006. 2010 - 2017.
- [6] Wang H, Li H, Liu Y, et al. Opposition-based particle swarm algorithm with cauchy mutation [A]. IEEE Congress on Evolutionary Computation [C]. Singapore: IEEE, 2007. 4750 - 4756.
- [7] Wang H, Wu Z J, Rahnamayan. Enhancing particle swarm optimization using generalized opposition-based learning [J]. Information Sciences, 2011, 181(20): 4699 - 4714.
- [8] 喻飞, 李元香, 等. 透镜成像反向学习策略在粒子群算法中的应用 [J]. 电子学报, 2014, 42(2): 230 - 235.
Yu Fei, Li Yuan-xiang, et al. The application of a novel OBL based on lens imaging principle in PSO [J]. Acta Electronica Sinica, 2014, 42(2): 230 - 235. (in Chinese)
- [9] Griffiths, David J. Introduction to Electrodynamics (3rd Ed) [M]. Prentice Hall, 1998. 386 - 389.
- [10] 周新宇, 吴志健, 等. 一种精英反向学习的粒子群优化算法 [J]. 电子学报, 2013, 41(8): 1647 - 1652.
Zhou Xin-yu, Wu Zhi-jian, et al. Elite opposition-based particle swarm optimization [J]. Acta Electronica Sinica, 2013, 41(8): 1647 - 1652. (in Chinese)
- [11] Rosenbrock H H. An automatic method for finding the greatest or least value of a function [J]. Computer Journal, 1960, 3(3): 175 - 184.
- [12] 梁艳春, 周春光, 等. 基于遗传算法的 Rosenbrock 函数优化问题的研究 [J]. 软件学报, 1997, 8(9): 701 - 708.
Liang Yan-chun, Zhou Chun-guang, et al. Optimization of Rosenbrock's function based on genetic algorithms [J]. Journal of Software, 1997, 8(9): 701 - 708. (in Chinese)
- [13] He S, Wu Q H, Saunders J R. Group search optimizer: an optimization algorithm inspired by animal searching behavior [J]. IEEE Transactions on Evolutionary Computation, 2009, 13(5): 973 - 990.
- [14] Changyong Lee, Xin Yao. Evolutionary programming using mutations based on the levy probability distribution [J]. IEEE Transactions on Evolutionary Computation, 2004, 2(8): 1 - 13.
- [15] 王晖. 区域变换搜索的智能算法研究 [D]. 武汉: 武汉大学, 2011. 17 - 19.
Wang Hui. Research on intelligent algorithms based on region transformation search and its applications [D]. Wuhan: Wuhan University, 2011. 17 - 19. (in Chinese)

作者简介



邵鹏男, 1983年生, 武汉大学计算机学院博士研究生, 研究方向: 智能计算、智能信息处理.

E-mail: sp198310@163.com



吴志健(通信作者) 男, 1963年生, 教授, 博士生导师, 武汉大学软件工程国家重点实验室副主任, 研究方向: 智能计算、并行计算和智能信息处理.

E-mail: zhijianwu@whu.edu.cn

周炫余 男, 1987年生, 武汉大学计算机学院博士研究生, 研究方向: 自然语言处理、智能信息处理.

邓长寿 男, 1972年生, 博士, 教授, 九江学院信息科学与技术学院副院长, 研究方向: 智能计算和数据挖掘.