

文章编号:1007-2985(2019)04-0024-09

惯性权值和学习因子对标准 PSO 算法性能的影响^{*}

宋梦培,莫礼平,周恺卿

(吉首大学信息科学与工程学院,湖南 吉首 416000)



摘 要:基于标准 PSO 算法,通过分析惯性权值和学习因子 2 类参数不同的取值策略对常用测试函数优化结果的影响,来探究 2 类参数对算法性能的影响.实验结果表明,2 类参数恰当的动态改变不仅能明显提高单峰函数的寻优精度和收敛速度,而且能提高双峰和多峰函数的寻优概率;惯性权值主要影响算法的收敛速度,随着惯性权值的递增,算法收敛速度逐渐加快;学习因子主要影响算法的寻优精度,当反映粒子的自我学习能力和向群体最优粒子学习的能力的学习因子同增同减变化时,寻优精度提高;惯性权值递增结合 2 种学习因子的同增同减变化,或惯性权值递减结合 2 种学习因子的一增一减变化,均可使标准 PSO 算法性能得到显著提高.

关键词:标准 PSO 算法;惯性权值;学习因子;收敛速度;寻优精度

中图分类号:TP301

文献标志码:A

DOI:10.13438/j.cnki.jdzk.2019.04.005

粒子群优化(Particle Swarm Optimization, PSO)算法,是 Kennedy 博士和 Eberhart 博士于 1995 年通过研究鸟群聚集和迁徙的觅食行为而提出的一种基于群体的自适应搜索优化技术^[1-2].由于 PSO 算法具有操作简单、易于实现和鲁棒性好等优点,因此该算法一经提出就引起了学者们的极大兴趣. SHI 等^[3]在粒子群算法中引入惯性因子,以平衡算法的全局搜索能力和局部搜索能力.这一改进算法被称为标准 PSO 算法. Suganthan^[4]提出了一种基于动态邻域的 PSO 模型.该模型中,用每个粒子的当前邻域极值代替全局极值,结合不同的领域结构有效地改进算法的收敛性. Higashi Natsuki 等^[5]在 PSO 算法中引入自适应的变异算子,从而提高了粒子跳出局部极值的能力和粒子的寻优概率. LI 等^[6]提出了一种能够增强粒子之间交流的动态学习策略,该策略能有效提高算法的运行效率. 张寅等^[7]提出了一种混沌策略,该策略通过在迭代停滞时初始化粒子位置来保证粒子的多样性. XIA 等^[8]通过定期对变量空间进行划分,逐步缩小搜索区域,来提高算法的收敛速度和寻优精度. Mahdavi 等^[9]提出了一种基于变量效应的协同多级进化策略,该策略通过决策变量将问题分解,减少了求解复杂度.改进后的粒子群算法能较大概率地找到问题的全局最优解,并且已成功应用于约束优化问题^[10-11]、调度问题^[12]、图形图像处理^[13]、数据挖掘^[14]和电力系统调控^[15]等领域.尽管学者们对标准 PSO 算法展开了广泛的讨论,但是关于惯性权值和学习因子对算法性能的影响并没有作详细探究.笔者拟选取惯性权值和学习因子 2 类参数,基于标准 PSO 算法,通过分析 2 类参数不同的取值策略对常用测试函数的优化结果的影响,来探究 2 类参数对算法性能的影响.

^{*} 收稿日期:2019-01-24

基金项目:国家自然科学基金资助项目(61462029);湖南省自然科学基金面上项目(2019JJ40234);吉首大学研究生科研项目(JDY1816)

通信作者:莫礼平(1972—),女,湖南安化人,高级实验师,主要从事智能计算及其应用、中文信息处理研究.

1 标准 PSO 算法的基本框架与描述

1.1 算法框架

标准 PSO 算法的基本框架如图 1 所示.

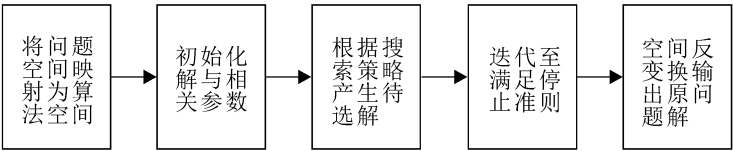


图 1 标准 PSO 算法的基本框架

Fig. 1 Basic Framework of the Standard PSO Algorithm

假设在一个 D 维的目标搜索空间中,有 N 个粒子组成一个群落,粒子自身最优位置为个体极值 p_{best} ,当前全局的最优位置为全局极值 g_{best} .每个粒子追随当前的最优粒子在解空间中运动,并根据如下公式来更新自己的速度和位置:

$$v_{i+1} = \omega v_i + c_1 r_1 (p_{best_i} - x_i) + c_2 r_2 (g_{best_i} - x_i), \tag{1}$$

$$x_{i+1} = x_i + v_{i+1}. \tag{2}$$

其中: ω 为惯性权值; c_1 和 c_2 为学习因子,分别反映粒子的自我学习能力和向群体最优粒子学习的能力; r_1 和 r_2 为 $[0,1]$ 的均匀随机数; v_i 为粒子速度, $v_i \in [-v_{max}, v_{max}]$, v_{max} 是用户设定的用来限制粒子速度的常量.

1.2 算法描述

标准 PSO 算法的流程如图 2 所示,其中 p_{recent} 为粒子当前的适应度值.

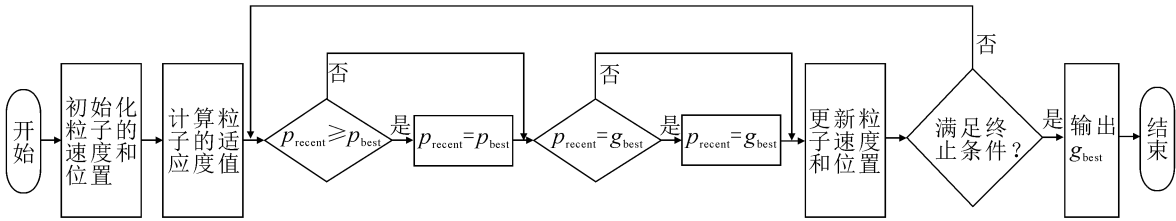


图 2 标准 PSO 算法的流程

Fig. 2 Flow Chart of the Standard PSO Algorithm

2 2 类参数对标准 PSO 算法的影响分析

由(1),(2)式可知,惯性权值 ω 和学习因子 c_1, c_2 的取值策略直接影响标准 PSO 算法的性能. ω 反映粒子对当前速度的继承情况:

- (i) 当 ω 较大时,粒子的全局搜索能力较强;
- (ii) 当 ω 较小时,粒子的局部搜索能力较强.

学习因子 c_1, c_2 分别反映粒子的自我学习能力和向群体最优粒子学习的能力:

- (i) 当 c_1 较大时,粒子的自我认知能力较强,容易偏离最优粒子;
- (ii) 当 c_2 较大时,粒子的社会认知能力较强,容易陷入局部最优.

笔者将基于标准 PSO 算法,通过分析 2 类参数不同的取值策略对常用测试函数优化结果的影响,来探究 ω, c_1, c_2 对标准 PSO 算法性能的影响.

3 测试实验

3.1 测试函数的选取

为了更好地评价改进参数后算法性能的优劣,选取 PSO 算法常用的 5 个测试函数 (Sphere, Rosenbrock, Griewank, Ackley, Rastrigrin) 进行实验设计,其中 Sphere 和 Rosenbrock 是单峰函数,其他均为多峰函数.测试函数的具体设置见表 1.

表 1 测试函数
Table 1 Test Function

函数名称	函数表达式	维数	搜索范围	目标值
Sphere	$f(x) = \sum_{i=1}^D x_i^2$	30	$[-30,30]$	0.01
Rosenbrock	$f(x) = \sum_{i=1}^{D-1} 100(x_{i+1} - x_i^2)^2 - (x_i - 1)^2$	30	$[-100,100]$	0.01
Griewank	$f(x) = \frac{1}{4\,000} \sum_{i=1}^D x_i^2 - \prod_{i=1}^D \cos \frac{x_i}{\sqrt{i}} + 1$	30	$[-600,600]$	0.1
Ackley	$f(x) = -20 \exp \left(-0.2 \sqrt{\frac{1}{D} \sum_{i=1}^D x_i^2} \right) - \exp \left(\frac{1}{D} \sum_{i=1}^D \cos(2\pi x_i) \right) + 20 + e$	30	$[-32,32]$	0.1
Rastrigrin	$f(x) = \sum_{i=1}^D (x_i^2 - 10 \cos(2\pi x_i) + 10)$	30	$[-10,10]$	0.1

3.2 参数变化策略的选取

在标准 PSO 算法中,惯性权值最优范围为 $[0.4,0.9]$,学习因子通常取常数 $2^{[3]}$. 为了探究 w, c_1, c_2 的动态变化对算法性能的影响,按如下策略进行取值:

惯性权值 w 的动态选取策略为:递增策略, $w = 0.4 + 0.5t/\text{MaxDT}$; 递减策略, $w = 0.9 - 0.5t/\text{MaxDT}$.

学习因子 c_1 和 c_2 的动态选取策略均为:递增策略, $c = 1 + \sin(t\pi/\text{MaxDT})$; 递减策略, $c = 2 - \sin(t\pi/\text{MaxDT})$. 其中: t 为迭代次数; MaxDT 为最大迭代次数.

3.3 实验结果与讨论

3.3.1 参数设置对 5 个测试函数均设计 15 组对比实验. 各组实验中,基本参数设置如下: $N=40, \text{MaxDT}=1\,000, D=5, w, c_1, c_2$ 的取值见表 2.

表 2 2 类参数的取值
Table 2 Value of the Two Types of Parameters

实验编号	参数	取值	实验编号	参数	取值
第 1 组	w	$w = 0.729\,8$	第 9 组	w 递增	$w = 0.4 + 0.5t/\text{MaxDT}$
	c_1	$c_1 = 1.496\,2$		c_1 递增	$c_1 = 1 + \sin(t\pi/\text{MaxDT})$
	c_2	$c_2 = 1.496\,2$		c_2 递增	$c_2 = 1 + \sin(t\pi/\text{MaxDT})$
第 2 组	w	$w = 0.729\,8$	第 10 组	w 递增	$w = 0.4 + 0.5t/\text{MaxDT}$
	c_1 递增	$c_1 = 1 + \sin(t\pi/\text{MaxDT})$		c_1 递减	$c_1 = 2 - \sin(t\pi/\text{MaxDT})$
	c_2 递减	$c_2 = 2 - \sin(t\pi/\text{MaxDT})$		c_2 递减	$c_2 = 2 - \sin(t\pi/\text{MaxDT})$
第 3 组	w	$w = 0.729\,8$	第 11 组	w 递减	$w = 0.9 - 0.5t/\text{MaxDT}$
	c_1 递减	$c_1 = 2 - \sin(t\pi/\text{MaxDT})$		c_1	$c_1 = 1.496\,2$
	c_2 递增	$c_2 = 1 + \sin(t\pi/\text{MaxDT})$		c_2	$c_2 = 1.496\,2$
第 4 组	w	$w = 0.729\,8$	第 12 组	w 递减	$w = 0.9 - 0.5t/\text{MaxDT}$
	c_1 递增	$c_1 = 1 + \sin(t\pi/\text{MaxDT})$		c_1 递增	$c_1 = 1 + \sin(t\pi/\text{MaxDT})$
	c_2 递增	$c_2 = 1 + \sin(t\pi/\text{MaxDT})$		c_2 递减	$c_2 = 2 - \sin(t\pi/\text{MaxDT})$
第 5 组	w	$w = 0.729\,8$	第 13 组	w 递减	$w = 0.9 - 0.5t/\text{MaxDT}$
	c_1 递减	$c_1 = 2 - \sin(t\pi/\text{MaxDT})$		c_1 递减	$c_1 = 2 - \sin(t\pi/\text{MaxDT})$
	c_2 递减	$c_2 = 2 - \sin(t\pi/\text{MaxDT})$		c_2 递增	$c_2 = 1 + \sin(t\pi/\text{MaxDT})$
第 6 组	w 递增	$w = 0.4 + 0.5t/\text{MaxDT}$	第 14 组	w 递减	$w = 0.9 - 0.5t/\text{MaxDT}$
	c_1	$c_1 = 1.496\,2$		c_1 递增	$c_1 = 1 + \sin(t\pi/\text{MaxDT})$
	c_2	$c_2 = 1.496\,2$		c_2 递增	$c_2 = 1 + \sin(t\pi/\text{MaxDT})$
第 7 组	w 递增	$w = 0.4 + 0.5t/\text{MaxDT}$	第 15 组	w 递减	$w = 0.9 - 0.5t/\text{MaxDT}$
	c_1 递增	$c_1 = 1 + \sin(t\pi/\text{MaxDT})$		c_1 递减	$c_1 = 2 - \sin(t\pi/\text{MaxDT})$
	c_2 递减	$c_2 = 2 - \sin(t\pi/\text{MaxDT})$		c_2 递减	$c_2 = 2 - \sin(t\pi/\text{MaxDT})$
第 8 组	w 递增	$w = 0.4 + 0.5t/\text{MaxDT}$			
	c_1 递减	$c_1 = 2 - \sin(t\pi/\text{MaxDT})$			
	c_2 递增	$c_2 = 1 + \sin(t\pi/\text{MaxDT})$			

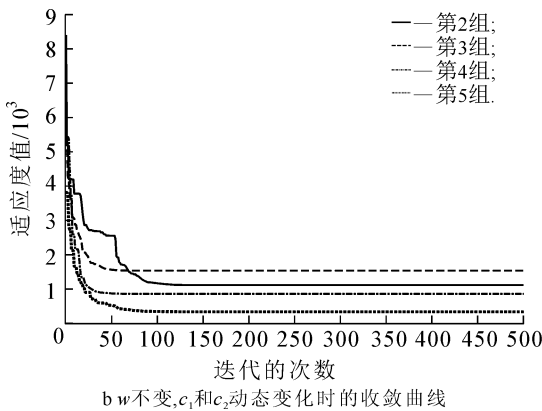
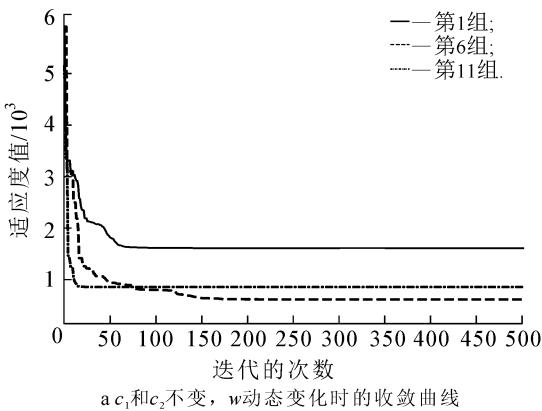
3.3.2 函数仿真结果 表 3 给出了 5 个测试函数的仿真结果,图 3~7 分别示出了 5 个测试函数的收敛曲线.

表 3 5 个测试函数的仿真结果

Table 3 Simulation Results of 5 Test Functions

实验编号	Sphere 函数最优解	Rosenbrock 函数最优解	Griewank 函数最优解
第 1 组	$6.039\ 664\ 173\ 406\ 676\times 10^{-82}$	3.930 839 434 133 028	0
第 2 组	$2.947\ 864\ 834\ 870\ 249\times 10^{-79}$	3.930 839 434 133 027	0
第 3 组	$4.477\ 289\ 762\ 976\ 436\times 10^{-78}$	$4.190\ 823\ 558\ 986\ 625\times 10^{-30}$	0
第 4 组	$5.331\ 212\ 423\ 031\ 509\times 10^{-69}$	$1.047\ 705\ 889\ 746\ 656\times 10^{-30}$	0
第 5 组	$1.217\ 441\ 528\ 398\ 872\times 10^{-88}$	3.930 839 434 133 027	0
第 6 组	$2.755\ 222\ 741\ 686\ 639\times 10^{-127}$	$3.109\ 654\ 696\ 448\ 099\times 10^{-15}$	0
第 7 组	$3.256\ 332\ 979\ 016\ 428\times 10^{-126}$	$1.220\ 459\ 145\ 360\ 667\times 10^{-10}$	0
第 8 组	$5.660\ 709\ 027\ 485\ 463\times 10^{-124}$	$1.426\ 186\ 117\ 195\ 466\times 10^{-24}$	0
第 9 组	$7.242\ 652\ 194\ 261\ 482\times 10^{-111}$	$1.265\ 710\ 578\ 854\ 400\times 10^{-23}$	0
第 10 组	$9.546\ 138\ 031\ 663\ 698\times 10^{-136}$	0	0
第 11 组	$2.165\ 556\ 367\ 379\ 605\times 10^{-123}$	3.930 839 434 133 028	0
第 12 组	$2.464\ 926\ 939\ 256\ 815\times 10^{-121}$	3.930 839 434 133 027	0
第 13 组	$6.073\ 714\ 827\ 872\ 535\times 10^{-121}$	$1.047\ 705\ 889\ 746\ 656\times 10^{-30}$	0
第 14 组	$1.234\ 372\ 426\ 973\ 556\times 10^{-108}$	3.930 839 434 133 027	0
第 15 组	$1.103\ 560\ 649\ 329\ 026\times 10^{-130}$	$1.047\ 705\ 889\ 746\ 656\times 10^{-30}$	0

实验编号	Ackley 函数最优解	Rastrigrin 函数最优解
第 1 组	$4.440\ 892\ 098\ 500\ 626\times 10^{-15}$	2.984 877 171 279 869
第 2 组	$4.440\ 892\ 098\ 500\ 626\times 10^{-15}$	5.969 749 304 740 667
第 3 组	$4.440\ 892\ 098\ 500\ 626\times 10^{-15}$	3.979 836 228 373 159
第 4 组	$4.440\ 892\ 098\ 500\ 626\times 10^{-15}$	3.979 836 228 373 161
第 5 组	$4.440\ 892\ 098\ 500\ 626\times 10^{-15}$	2.984 877 171 279 870
第 6 组	$4.440\ 892\ 098\ 500\ 626\times 10^{-15}$	1.989 918 114 186 580
第 7 组	$4.440\ 892\ 098\ 500\ 626\times 10^{-15}$	3.979 836 228 373 161
第 8 组	$4.440\ 892\ 098\ 500\ 626\times 10^{-15}$	4.974 790 247 647 375
第 9 组	$4.440\ 892\ 098\ 500\ 626\times 10^{-15}$	4.974 790 247 647 377
第 10 组	$7.993\ 605\ 777\ 301\ 127\times 10^{-15}$	3.979 836 228 373 160
第 11 组	$4.440\ 892\ 098\ 500\ 626\times 10^{-15}$	3.979 836 228 373 163
第 12 组	$4.440\ 892\ 098\ 500\ 626\times 10^{-15}$	3.979 836 228 373 161
第 13 组	$7.993\ 605\ 777\ 301\ 127\times 10^{-15}$	3.979 836 228 373 161
第 14 组	$8.881\ 784\ 197\ 001\ 252\times 10^{-16}$	3.979 836 228 373 160
第 15 组	$8.881\ 784\ 197\ 001\ 252\times 10^{-16}$	5.969 749 304 740 665



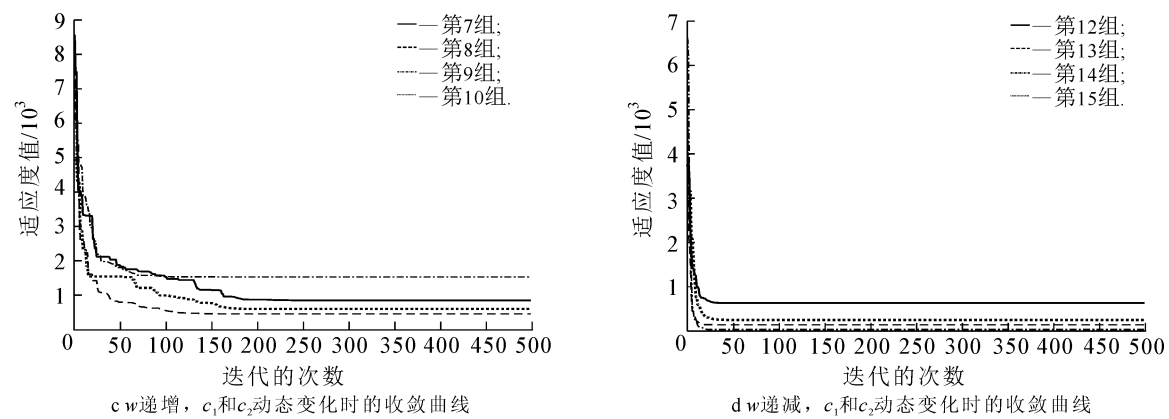


Fig. 3 Convergence Curve of Sphere Function

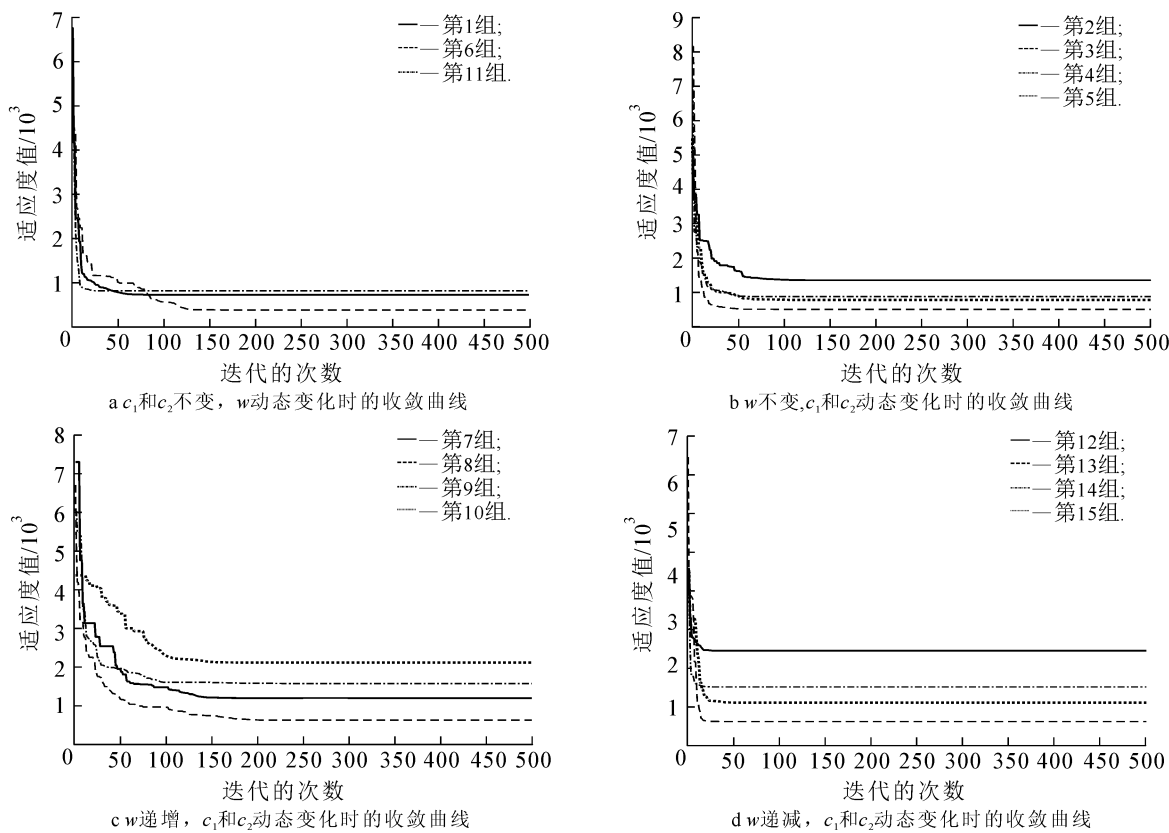
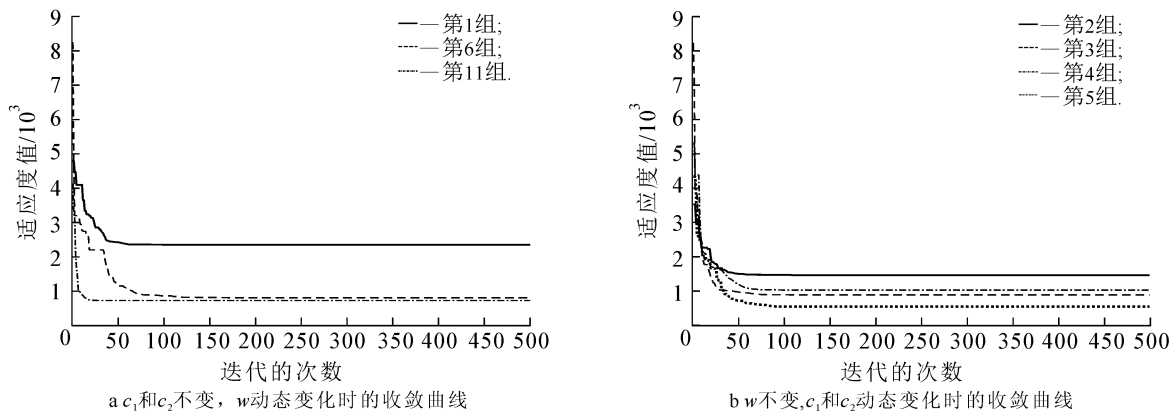


Fig. 4 Convergence Curve of Rosenbrock Function



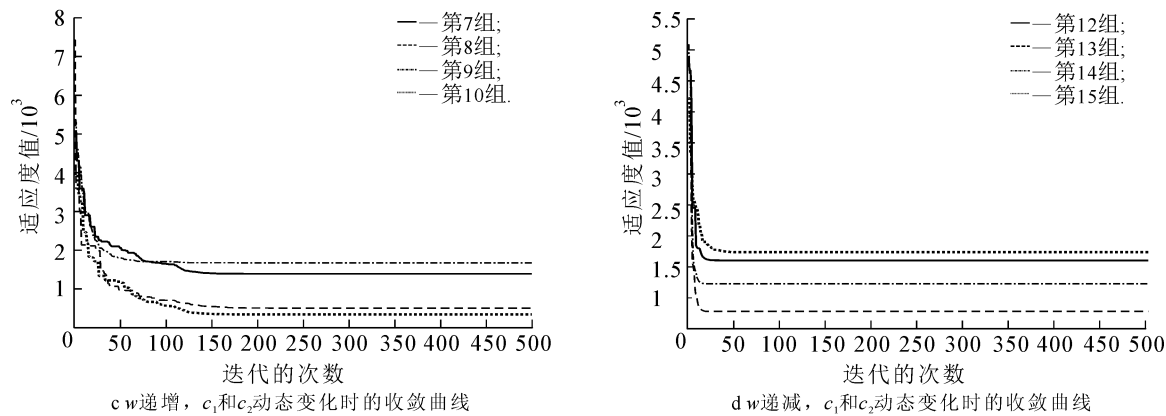


图 5 Griewank 函数的收敛曲线

Fig. 5 Convergence Curve of Griewank Function

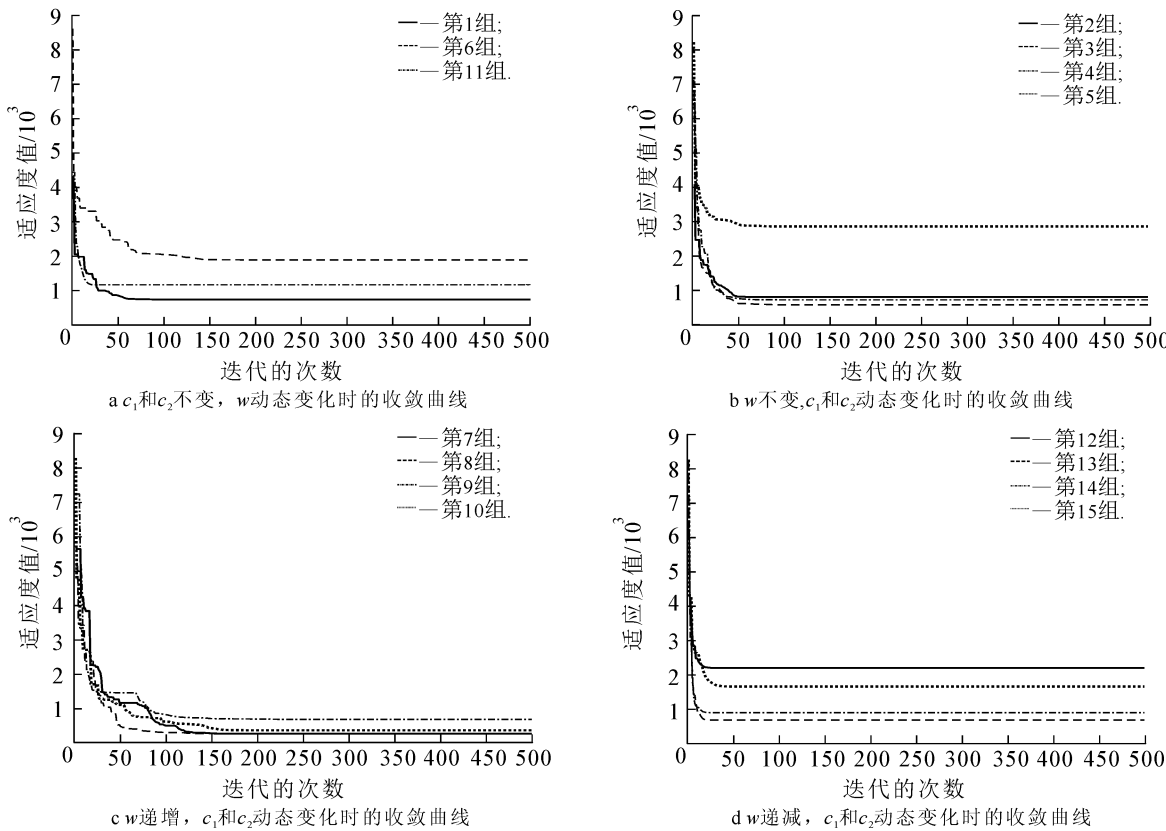
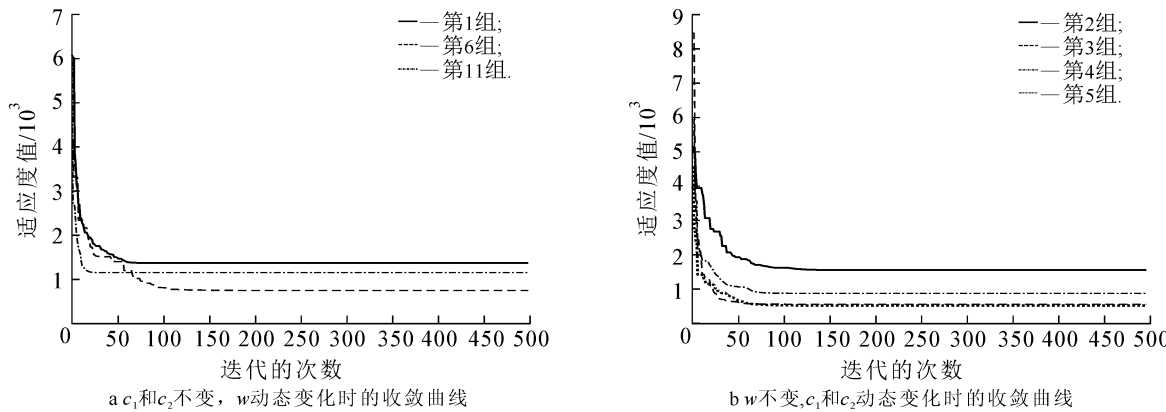


图 6 Ackley 函数的收敛曲线

Fig. 6 Convergence Curve of Ackley Function



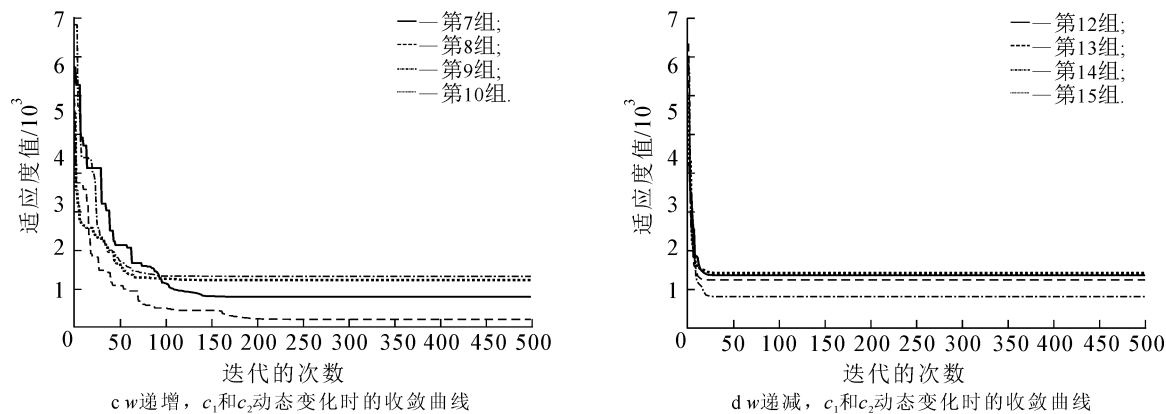


图 7 Rastrigrin 函数的收敛曲线

Fig. 7 Convergence Curve of Rastrigrin Function

Sphere 函数是只有全局极值点而没有局部极值点的单峰函数,通常用来测试算法的优化精度.由表 3 和图 3 可知:

- (1) 当 c_1, c_2 为常数且 w 采取动态改变策略时,标准 PSO 算法的寻优精度明显提高.当 w 递增时,收敛速度加快;当 w 递减时,寻优精度提高.
- (2) 当 w 为常数且 c_1, c_2 采取动态改变策略时,标准 PSO 算法的寻优精度显著提高,且比动态改变 w 时的精度更高.当 c_2 递增时,收敛速度加快.
- (3) 当 w 递增时,标准 PSO 算法的收敛速度明显加快.当 c_1, c_2 同增同减时,收敛适应度值降低;当 c_2 递增时,寻优精度明显提高.
- (4) 当 w 递减时,标准 PSO 算法的收敛速度略微降低.但当 c_1 递减时,收敛适应度值明显降低.

Rosenbrock 函数是只有全局极值点而没有局部极值点的非凸病态函数,极难收敛于全局极值点,通常用来评价优化算法的执行效率.由表 3 和图 4 可知:

- (1) 当 c_1, c_2 为常数且 w 递增时,标准 PSO 算法的收敛速度加快,但寻优概率并没有明显提高;当 c_1, c_2 为常数且 w 递减时,寻优概率略微提高,寻优波动的范围明显缩小.
- (2) 当 w 为常数且 c_1, c_2 同增同减时,标准 PSO 算法的收敛速度加快;当 w 为常数且 c_1, c_2 一增一减时,收敛适应度值降低. c_1, c_2 的动态改变缩小了寻优波动的范围,提高了粒子寻优概率.
- (3) 当 w 递增时,标准 PSO 算法的收敛速度明显加快.当 c_1 递减时,收敛适应度值降低;当 c_1, c_2 同增同减时,寻优概率提高.
- (4) 当 w 递减时,标准 PSO 算法的收敛速度略微降低.但当 c_1, c_2 同增同减时,收敛适应度值降低,算法的寻优概率提高.

Griewank 函数是由多个局部极值点包围 1 个全局极值点的旋转多峰函数,在 5 维空间均能找到全局最优解.由表 3 和图 5 可知:

- (1) 当 c_1, c_2 为常数且 w 递增时,标准 PSO 算法的收敛速度加快;当 c_1, c_2 为常数且 w 递减时,寻优概率提高.
- (2) 当 w 为常数且 c_1, c_2 同增同减时,标准 PSO 算法的收敛速度加快,收敛适应度值降低.当 c_2 递减时,寻优概率提高;当 c_2 递增时,寻优精度提高.
- (3) 当 w 递增时,标准 PSO 算法的收敛速度明显加快,且 c_1, c_2 一增一减时,收敛适应度值降低.当 c_2 递减时,寻优概率提高;当 c_2 递增时,寻优精度提高.
- (4) 当 w 递减且 c_1, c_2 同增同减时,标准 PSO 算法的寻优概率和寻优精度同步提高,收敛适应度值降低.

Ackley 函数是具有很多局部极值的多峰函数,其图像是一个有很多孔峰、起伏不平的曲面, w, c_1, c_2 的动态改变在 5 维均能找到最优解.由表 3 和图 6 可知:

- (1) 当 c_1, c_2 为常数且 w 递增时,标准 PSO 算法的收敛速度加快;当 c_1, c_2 为常数且 w 递减时,寻优精

度提高.

- (2) 当 ω 为常数且 c_2 递增时,标准 PSO 算法的收敛速度和寻优精度同步提高.
- (3) 当 ω 递增时,标准 PSO 算法的收敛速度明显加快,且当 c_1 递增时,寻优概率和寻优精度同步提高.
- (4) 当 ω 递减且 c_1, c_2 同增同减时,收敛适应度值降低,但收敛速度和寻优概率并没有明显改变.

Rastrigrin 函数是典型的多峰函数,只有 1 个被多个随余弦函数波动的局部极值包围的全局极值,很难找到全局极值点.由表 3 和图 7 可知:

- (1) 动态改变 ω, c_1, c_2 ,只能适当缩小标准 PSO 算法寻优波动的范围,而不能找到全局的最优解.但当 c_1, c_2 为常数且 ω 递增时,收敛速度明显加快;当 c_1, c_2 为常数且 ω 递减时,寻优波动范围缩小.
- (2) 当 ω 为常数且 c_2 递增时,标准 PSO 算法的收敛速度加快;当 ω 为常数且 c_2 递减时,寻优波动范围缩小.
- (3) 当 ω 递增时,标准 PSO 算法的收敛速度明显加快,且当 c_1, c_2 同增同减时,收敛适应度值降低.
- (4) 当 ω 递减时,标准 PSO 算法的收敛速度略微降低,但当 c_1, c_2 一增一减时,收敛适应度值明显降低.

4 结语

基于标准 PSO 算法,通过分析惯性权值和学习因子 2 类参数不同的取值策略对常用测试函数优化结果的影响,来探究 2 类参数对算法性能的影响.实验结果表明:

- (1) ω 主要影响标准 PSO 算法的收敛速度,随着 ω 的递增,收敛速度明显加快; c_1, c_2 主要影响寻优精度,随着 c_2 的递增,寻优精度提高.
- (2) ω 递减与 c_1, c_2 一增一减结合,对标准 PSO 算法性能的优化较好; ω 递增与 c_1, c_2 同增同减结合,对算法性能的优化效果较好.
- (3) 当 ω, c_1, c_2 均为动态函数时,单峰函数的寻优精度和收敛速度明显提高,双峰或多峰函数波动的范围缩小且寻优概率提高.
- (4) ω, c_1, c_2 的动态改变对函数低维优化效果明显,但随着维数的增加,寻优难度加大,优化效果迅速降低.

考虑到惯性权值和学习因子的改变仅对单峰函数的优化效果较明显,对多峰或双峰函数并没有明显的优化作用,下一步将结合蚁群算法、遗传算法等来探究 2 类参数对函数优化效果的影响.

参考文献:

[1] KENNEDY J, EBERHART R C. Particle Swarm Optimization[C]. Proceedings of IEEE International Conference on Neural Networks. Perth, Australia:IEEE Service Center, 1995: 1 942 - 1 948.

[2] EBERHART R C, SHI Yuhui. Comparing Inertia Weights and Constriction Factors in Particle Swarm Optimization [C]. Proceedings of the Congress on Evolutionary Computation Nagoya, Japan: IEEE Press, 2000: 84 - 88.

[3] SHI Yuhui, EBERHART R A. Modified Particle Swarm Optimizer[C]. IEEE World Congress On Computational Intelligence. Washington DC: IEEE Press, 1998: 69 - 73.

[4] SUGANTHAN P N. Particle Swarm Optimizer with Neighborhood Operator[C]. Proceedings of the Congress on Evolutionary Computation. Washington DC, Piscataway, NJ: IEEE Service Center, 1999: 1 958 - 1 962.

[5] NATSUKI HIGASHI, HITOSHI IBA. Particle Swarm Optimization with Gaussian Mutation[C]. Proceedings of the Swarm Intelligence Symposium, Indianapolis: IEEE Service Center, 2003: 72 - 79.

[6] LI Changhe, YANG Shengxiang. An Adaptive Learning Particle Swarm Optimizer for Function Optimization[C]. IEEE Congress on Evolutionary Computation, Trondheim: IEEE Service Center, 2009: 381 - 388.

[7] 张 寅,曹德欣.混沌动态种群数粒子群优化算法[J]. 计算机工程与应用,2011,47(35):38 - 40.

[8] XIA Xuewen, LIU Jingnan, HU Zhongbo. An Improved Particle Swarm Optimizer Based on Tabu Detecting and Local Learning Strategy in a Shrunk Search Space Applied[J]. Soft Computing, 2014, 23(1): 76 - 90.

[9] MAHDAVI SEDIGHEH, RAHNAMAYAN SHAHRYAR. Shiri Mohammad Ebrahim Multilevel Framework for

Large-Scale Global Optimization[J]. *Soft Computing*, 2016, 21(14): 4 111 – 4 140.

[10] CAMPOS MAURO, KROHLING RENATO A. Hierarchical Bare Bones Particle Swarm for Solving Constrained Optimization Problems[C]. *Proceedings of the IEEE Congress on Evolutionary Computation*. Cancun, Mexico: IEEE Press, 2013: 805 – 812.

[11] LI Jie, ZHANG Junqi, JIANG Changjun, et al. Composite Particle Swarm Optimizer with Historical Memory for Function Optimization[J]. *IEEE Transactions on Cybernetics*, 2015, 45(10): 2 350 – 2 363.

[12] SHOU Yongyi, LI Ying, LAI Changtao. Hybrid Particle Swarm Optimization for Preemptive Resource-Constrained Project Scheduling[J]. *Neurocomputing*, 2015, 148: 122 – 128.

[13] 赵玉磊,郭宝龙,吴宪祥,等. 基于双粒子群协同优化的 ECT 图像重建算法[J]. *计算机研究与发展*, 2014, 51(9): 2 094 – 2 100.

[14] ZHANG Yong, GONG Dunwei, HU Ying, et al. Feature Selection Algorithm Based on Bare Bones Particle Swarm Optimization[J]. *Neurocomputing*, 2015, 148: 150 – 157.

[15] KOSHITI A, ARYA L D, CHOUBE S C. Voltage Stability Constrained Distributed Generation Planning Using Modified Bare Bones Particle Swarm Optimization[J]. *Journal of the Institution of Engineers(India): Series B*, 2013, 94(2): 123 – 133.

Influence of Inertia Weight and Learning Factor on Performance of Standard PSO Algorithm

SONG Mongpei, MO Liping, ZHOU Kaiqing

(College of Information Science and Engineering, Jishou University, Jishou 416000, Hunan China)

Abstract: Two types of parameters, inertia weight and learning factor, are the main factors affecting the accuracy and convergence speed of the standard PSO algorithm. Based on the standard PSO algorithm, the influence of different value-taking strategies of two types of parameters on the optimization results of commonly used test functions is analyzed to explore its impact on the performance of the standard PSO algorithm. The experimental results show that the proper dynamic change of the two types of parameters can significantly improve the optimization accuracy and convergence speed of the unimodal function, and can improve the optimization probability of the bimodal and multimodal functions. The inertia weight mainly affects the convergence speed of the algorithm. As the inertia weight increases, the convergence speed of the algorithm increases. The learning factor mainly affects the optimization precision of the algorithm. When the two learning factors increase and decrease, the precision of the optimization increases. The inertia weight increases with the two learning factors. The increase and decrease of the change, or the decrease of the inertia weight combined with the increase and decrease of the two learning factors, can significantly improve the performance of the standard PSO algorithm.

Key words: standard particle swarm optimization; inertia weight; learning factors; convergence speed; optimization accuracy

(责任编辑 向阳洁)