

一种结合灰狼算法的粒子群优化算法

李真^{1,2,3}, 王帆^{4,5}, 王冉珺^{1,3}

- (1. 中国科学院光电技术研究所, 成都 610209; 2. 中国科学院大学, 北京 100049;
3. 中国科学院空间光电精密测量技术重点实验室, 成都 610209;
4. 中国科学院西安光学精密机械研究所, 西安 710000;
5. 中国科学院空间精密测量技术重点实验室, 西安 710000)

摘要: 针对粒子群算法有陷入局部最优的缺点, 提出一种基于灰狼算法的粒子群优化算法。首先, 根据自然界中优胜劣汰的生存法则, 对每次迭代种群中的最差粒子进行进化, 其次, 由于粒子群算法中整个种群中的最优粒子有很强的引导能力, 对最优粒子进行扰动, 增大寻找全局最优的可能性; 最后, 结合灰狼优化算法, 引导粒子群包围式进行搜索, 增强全局搜索能力; 将改进的粒子群算法与标准粒子群算法在 9 个测试函数上进行了寻优精度和收敛速度的对比, 结果证明改进粒子群算法 (PSO_GWO) 在收敛速度和寻优精度上均优于粒子群算法 (PSO)。

关键词: 粒子群算法; 最差粒子; 最优粒子; 灰狼算法

An Improved Particle Swarm Optimization Algorithm

LI Zhen^{1,2,3}, WANG Fan^{4,5}, WANG Ranjun^{1,3}

- (1. Institute of Optics and Electronics of the Chinese Academy of Sciences, Chengdu 610209, China;
2. University of Chinese Academy of Sciences, Beijing 100049, China;
3. Key Laboratory of Science and Technology on Space Optoelectronic Precision Measurement, Chinese Academy of Sciences, Chengdu 610209, China;

4. Xi'an Institute of Optics and Precision Machinery, Chinese Academy of Sciences, Xi'an 710000, China;
5. Key Laboratory of Space Precision and Measurement Technology, Chinese Academy of Sciences, Xi'an 710000, China)

Abstract: Aiming at the shortcoming of particle swarm optimization (PSO) falling into local optimum, a particle swarm optimization algorithm based on Gray Wolf algorithm is proposed. Firstly, according to the survival law of the fittest in nature, the worst particles in each iteration population are evolved. Secondly, because the optimal particles in the whole population have strong guiding ability, they disturb the optimal particles and increase the possibility of finding the global optimal. Finally, combined with the gray wolf optimization algorithm, the particle swarm optimization algorithm is guided to search in an encircled manner to enhance the global search ability. The results show that the improved PSO is better than PSO in convergence speed and optimization accuracy.

Keywords: PSO; worst particle; best particle; GWO

0 引言

群体智能算法^[1]如蚁群算法^[2]、鲨鱼算法^[3]、蜂群算法^[4-5]、遗传算法^[6-8]、粒子群算法^[9-11]近几年逐渐成为学者们研究的焦点, 由于粒子群算法 (PSO, particle swarm optimization) 易于实现且复杂度相对较小, 已广泛应用于多个不同的工程领域^[12-13], 但粒子群算法存在一个天然缺陷, 就是其在搜索的过程当中, 易陷入局部最优, 即最终通过粒子群算法找到的值与全局最优值有一定的差距, 针对此缺点许多学者提出了许多改进策略。文献 [14] 将提出将遗传算法与粒子群算法结合使算法能够快速收敛, 文献 [15] 将粒子群算法的位置更新公式进行改进, 增强其搜索能力, 文献 [16] 将混沌策略加入粒子群算法当中, 改善

了其易陷入局部最优的缺点。文献 [17] 对粒子群算法的参数进行调整, 平衡全局和局部的搜索能力。

本文同样针对粒子群算法易陷入局部最优的缺点, 对粒子群算法进行改进, 由此提出了一种改进的粒子群优化算法 (PSO_GWO), 主要改进策略包括最差粒子进行进化策略和对最优粒子进行扰动的策略, 以及结合灰狼算法搜索能力强^[18]的优点, 将灰狼算法的优势引入粒子群算法中, 并且选取了 9 个常用的标准测试函数进行测试, 从收敛速度和寻优精度两个方面进行了对比, 仿真结果表明所提算法有效。

1 粒子群算法

粒子群算法 (PSO) 是 1995 年 Kennedy^[11]提出的通过

收稿日期: 2021-03-09; 修回日期: 2021-04-07。

基金项目: 国家自然科学基金青年项目 (61905220); 国家级大学生创新创业训练计划项目 (202010338014)。

作者简介: 李真 (1995-), 女, 河南省驻马店市人, 硕士, 主要从事控制器参数自整定方向的研究。

引用格式: 李真, 王帆, 王冉珺, 等. 一种结合灰狼算法的粒子群优化算法[J]. 计算机测量与控制, 2021, 29(10): 217-222.

仿照鸟群寻找食物的过程的一种进化算法，其基本思想是将粒子看作没有体积和质量的个体，每个粒子都有速度和位置两个特征，每个粒子在每次迭代过程中朝着粒子群全局最优和自身历史最优趋近，不断进化寻找更优值。粒子群算法迭代更新公式如下：

$$v_{i,j}^{t+1} = \omega v_{i,j}^t + c_1 r_1 (p_{best,i} - x_{i,j}^t) + c_2 r_2 (g_{best} - x_{i,j}^t) \quad (1)$$

$$x_{i,j}^{t+1} = x_{i,j}^t + v_{i,j}^{t+1} \quad (2)$$

其中： ω 是惯性权重因子； $v_{i,j}^{t+1}$ 指第 t 次迭代的第 i 个粒子的第 j 维的速度； $x_{i,j}^{t+1}$ 指第 t 次迭代的第 i 个粒子的第 j 维的位置； p_{best} 指粒子个体历史最优位置； g_{best} 指粒子群全局最优位置； c_1, c_2 分别为学习因子， r_1, r_2 为分布在 $[0, 1]$ 上的随机数。

2 粒子群优化策略

由于粒子群算法有易陷入局部最优的缺点，其陷入局部最优的原因主要有两点：第一点是由于粒子群算法在迭代后期会发生聚集造成种群多样性下降；第二点是在算法迭代更新公式中只考虑了个体经验和最优粒子的引导作用，同样会导致算法出现多样性下降和早熟收敛的问题。所以，本文为提高粒子群算法的搜索能力，提出了改进粒子群算法 (PSO_GWO)，分别提出了最差改进策略、最优粒子扰动策略，并结合灰狼算法对迭代更新公式进行改进，其中最差粒子改进策略是为了提高粒子群算法在搜索过程中的多样性，而最优粒子扰动策略则是为了防止算法后期发生聚集导致搜索能力下降，灰狼算法的引进是为了改进现有粒子群算更新公式中所存在的缺点。

2.1 最差粒子改进策略

由于粒子群算法是一种模仿自然界中鸟群觅食的过程提出的一种智能优化算法，受自然界中“优胜劣汰”生存法则的启发，对粒子群算法中的进行优化，即在每次迭代过程中对适应度最差的粒子进行进化，其主要目的是为了增加粒子群的多样性增强算法的搜索能力，如果进化后的粒子的适应度优于未进化时的适应度，则将进化后的粒子替换掉未进化的粒子，如果进化后的粒子的适应度比未进化时的适应度差，则保留未进化时的粒子。对最差适应度的粒子进化策略具体如下：

$$x_worst(t) = \text{argmax}(fit(x_1(t)), fit(x_2(t)) \dots fit(x_N(t))) \quad (3)$$

$$Eworst(t) = x_i(t) + \text{rand}(x_j(t) + x_k(t)) \quad (4)$$

($i \neq j \neq k \in 1, \dots, N$)

$$x_worst(t) = \begin{cases} Eworst(t) & \text{if}(fit(Eworst) < fit(x_worst)) \\ x_worst(t) & \text{otherwise} \end{cases} \quad (5)$$

式中， $x_worst(t)$ 表示第 t 次迭代时， N 个粒子中最差适应度的粒子，其中 N 表示粒子群的数量为 N 。式 (4) 中 $Eworst(t)$ 表示进化后的粒子的位置，其中 i, j, k 为在粒子种群中随机选择的 3 个互不相同的粒子。如式 (5) 所示，若进化后的粒子的适应度优于未进化的粒子的适应度，则

替换掉进化前的粒子。若进化后的粒子的适应度比未进化的粒子的适应度差，则保留原有粒子。

2.2 最优粒子扰动策略

在算法迭代后期，由于粒子群发生聚集导致粒子群的搜索能力下降，从而使得粒子群算法极易陷入局部最优。粒子群算法的全局最优位置 $g_{best}(t)$ 在整个粒子群算法寻优过程中对种群有很强的引导作用，全局最优位置会引导所有粒子朝着全局最优位置方向进行，所以可以根据粒子群的聚集程度，对全局最优位置进行扰动，从而增大粒子群的全局搜索能力。

$$avg_x(j) = \frac{\sum_{i=1}^N x[i][j]}{N} \quad (6)$$

$$r_i = |x_i - avg_x| \quad (7)$$

$$avg_r = \frac{\sum_{i=1}^N r_i}{N} \quad (8)$$

式中， $avg_x(j)$ 表示粒子群中所有粒子的中心位置的第 j 维， i 表示第 i 个粒子。式 (7) 中 r_i 表示第 i 个粒子与粒子群中心位置的距离，式 (8) 表示所有粒子到中心位置的平均距离。若 avg_r 较大说明粒子群较为分散，即说明粒子群还具有很强的多样性，此时只需要对全局最优位置进行较小的扰动，即增强粒子群的局部搜索能力。若 avg_r 较小说明粒子群较为密集，此时粒子群的多样下降，甚至粒子群可能已经陷入局部最优，此时需要对全局最优位置进行较大的扰动，使得全局最优位置引导粒子群跳出局部最优，即增强粒子群全局搜索能力。对全局最优位置的扰动策略如式 (9)，使用指数函数对全局最优粒子进行扰动， avg_r 较小时，对应的指数函数值较大，即产生的扰动较大。反之，对应的指数函数值较小，即产生的扰动较小。

$$N_{best}(t) = \text{sign}(\text{rand} - 0.5) \exp(-avg_r) \quad (9)$$

$$g_{best}(t) + g_{best}(t)$$

$$G_{best}(t) = \begin{cases} N_{best}(t) & \text{if}(fit(N_{best}(t)) < fit(G_{best}(t))) \\ G_{best}(t) & \text{otherwise} \end{cases} \quad (10)$$

式中，若扰动后的粒子的适应值优于扰动前的粒子，则将扰动前的粒子替换为扰动后的粒子；反之，扰动前的粒子将不进行替换。

2.3 灰狼包围优化策略

最差粒子进化策略提高了粒子群算法的多样性，最优粒子进化策略帮助算法在迭代后期跳出局部最优，针对粒子群算法迭代更新公式中只考虑了个体经验和最优粒子的引导作用，导致算法出现多样性下降和早熟收敛的问题。本文结合灰狼算法对其更新公式进行改进，主要原因是灰狼算法在迭代更新时，采用精英群体引导策略，即选择群体中最优的 3 个精英个体进行引导更新，而不再是粒子群算法中最优的一个个体进行引导。而且灰狼算法采用的是

包围式引导策略,即在搜索过程中,其他粒子会包围式的向精英群体进行趋近。所以将粒子群算法与狼群算法进行结合可以进一步提高算法的搜索能力。

灰狼算法中最重要的搜索策略的包围式搜索策略,其数学模型如下。

$$\mathbf{X}(t+1) = \mathbf{X}_p(t) - A \times D \quad (11)$$

$$D = |C \times \mathbf{X}_p(t) - \mathbf{X}(t)| \quad (12)$$

其中: t 为迭代次数, $\mathbf{X}_p(t)$ 是猎物的位置向量, $\mathbf{X}(t)$ 是灰狼的位置向量。其包围示意图如图 1 所示。

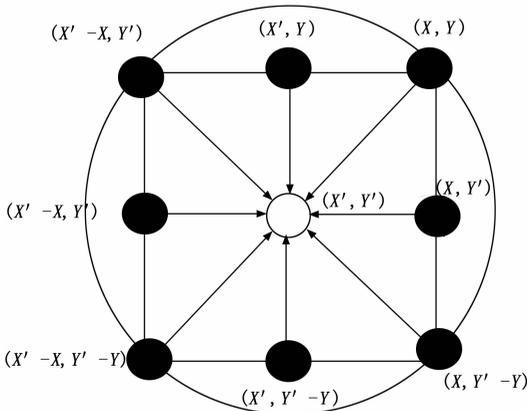


图 1 灰狼算法包围策略示意图

如图 1 所示,假设猎物位置为 (X^*, Y^*) ,灰狼位置为 (X, Y) ,在 $\vec{A} = (1, 0)$ 、 $\vec{C} = (1, 1)$ 时,灰狼会根据猎物位置根据公式移动到 $(X^* - X, Y^*)$,所以在向量 \vec{A} 和 \vec{C} 所取值不同时,会产生如图 1 的包围的效果。其中 \vec{A} 和 \vec{C} 为系数向量,可由式 (13) 和 (14) 表示:

$$C = 2r_1 \quad (13)$$

$$A = 2ar_2 - a \quad (14)$$

其中: r_1 、 r_2 为 $[0, 1]$ 之间的随机数, a 为控制参数在 $[0, 2]$ 内随着迭代次数线性递减,递减公式如式 (15):

$$a = 2(1 - \frac{t}{T_{\max}}) \quad (15)$$

由上述对灰狼算法的介绍中可知,灰狼算法在进行引导时,其采用精英群体对种群进行引导,即选择最优的 3 个精英个体对种群进行引导。灰狼种群中最优的 3 个精英个体分别是最优解 α 、次优解 β 以及第三最优解,3 个精英个体均会根据 (11)、(12) 对整个种群进行包围式的引导,其引导策略具体公式如下:

$$D_\alpha = |C_1 \times X_\alpha(t) - X(t)| \quad (16)$$

$$D_\beta = |C_1 \times X_\beta(t) - X(t)| \quad (17)$$

$$D_\delta = |C_1 \times X_\delta(t) - X(t)| \quad (18)$$

其中:式 (16) ~ (18) 为群体中灰狼个体分别距离前 3 个最优解 α 、 β 、 δ 的距离,式 (19) ~ (21) 为灰狼种群分别向 3 个精英个体移动的方向,式 (22) 为结合 3 个精英个体引导后,灰狼算法的更新公式:

$$X_1(t) = X_\alpha(t) - A_1 \times D_\alpha \quad (19)$$

$$X_2(t) = X_\beta(t) - A_2 \times D_\beta \quad (20)$$

$$X_3(t) = X_\delta(t) - A_3 \times D_\delta \quad (21)$$

$$X(t+1) = \frac{X_1(t) + X_2(t) + X_3(t)}{3} \quad (22)$$

为引入灰狼算法包围式搜索的全局搜索能力强的优点,将粒子群算法的更新公式改为式 (23),即在粒子群算法更新公式中添加灰狼算法的包围式更新策略:

$$v_{i,j}^{t+1} = \omega(X_{i,j}^t - x_{i,j}^t) + c_1 r_1 (p_{best_{i,j}}^t - x_{i,j}^t) + c_2 r_2 (g_{best}^t - x_{i,j}^t) \quad (23)$$

$$x_{i,j}^{t+1} = x_{i,j}^t + v_{i,j}^{t+1}$$

通过式 (23) 可以看出,在粒子群算法的位置更新过程中,保留了粒子群算法原本的个体经验和最优粒子的引导,并且引入了灰狼算法更新策略 $X(t)$,其不仅包含了灰狼算法的精英群体的引导策略还包含了灰狼算法的包围式搜索策略。通过引入灰狼算法,可以弥补粒子群算法在迭代更新公式中存在的不足。

3 仿真分析

为验证算法的有效性,采用表 1 中的 9 个测试函数进行测试,这 9 个测试函数表示不同的参数空间并且拥有不同的特性,可以充分考察算法的寻优能力,函数表达式和函数的搜索范围以及 9 个函数的全局最优值如表 1 所示,其中 $f_1 \sim f_3$ 、 $f_6 \sim f_9$ 为单峰函数,表示只有一个最小值; $f_4 \sim f_5$ 为多峰函数,表示有多个局部最小值,将优化算法在不同函数上进行寻优,测试算法性能。

3.1 算法寻优精度对比分析

将 PSO、GWO、GAPSO、PSOd 以及本文提出的改进粒子群算法 (PSO_GWO) 分别对测试函数寻优,每种算法的种群规模均为 $N=20$,算法的迭代次数均为 $T_{\max}=1000$,对每种算法重复运行 20 次,对 20 次的结果求均值 (mean) 和方差 (std) 进行对比,最终对 5 种不同算法的性能进行了排名 (rank),5 种算法的寻优精度对比以及最终的结果排名如表 2 所示。

由表 2 可知,将本文提出的改进粒子群算法 (PSO_GWO) 与其他 4 种算法做对比时,函数 f_1 、 f_2 、 f_4 、 f_5 、 f_6 、 f_8 均可找到全局最优值 0,而标准粒子群 (PSO) 和改进位置更新公式的粒子群算法 (PSO_d) 在对这 6 个函数进行寻优时,均未找到全局最优值 0,并且最终找到的值与全局最优值 0 差距较大。对于 f_7 函数,对比其他 4 种算法,PSO_GWO 的寻优精度与 GAPSO 的寻优精度相同,但是比较其他 3 种算法其寻优精度提高了 1~16 个数量级。对 f_3 函数进行寻优时,PSO_GWO 的寻优精度比较其他 4 种算法寻优精度提高了 1~4 个数量级。对 f_9 函数进行寻优时,PSO_GWO 的寻优精度比较其他 4 种算法寻优精度提高了 1~4 个数量级,综上所述,PSO_GWO 在对不同测试函数上进行寻优时,只有在 f_7 函数上进行寻优时,寻优结果与 GAPSO,在其余函数上进行寻优时的寻优结果均是优于其他 4 种算法的,所以在对 5 种算法进行综合评价时,PSO_GWO 的综合排名是处于第 1 位的,即 PSO_GWO 在寻优精度上是优于其余 4 种算法的。

表 1 9 个不同的测试函数函数

名称	函数	搜索范围	全局最优值
Sphere	$f_1(x) = \sum_{i=1}^D x_i^2$	$[-100, 100]$	0
Step	$f_2(x) = \sum_{i=1}^D (x_i + 0.5)^2$	$[-100, 100]$	0
Rosenbrock	$f_3(x) = \sum_{i=1}^D [100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2]$	$[-32, 32]$	0
Griewank	$f_4(x) = \frac{1}{4000} \sum_{i=1}^D x_i^2 + \prod_{i=1}^D \cos(\frac{x_i}{\sqrt{i}}) + 1$	$[-600, 600]$	0
Rastrigin	$f_5(x) = \sum_{i=1}^D [x_i^2 - 10\cos(2\pi x_i) + 10]$	$[-32, 32]$	0
Schwefel's Problem1.2	$f_6(x) = \sum_{i=1}^D (\sum_{j=1}^i x_j)^2$	$[-100, 100]$	0
Ackley	$f_7(x) = -20 \exp(-0.2 \sqrt{\frac{\sum_{i=1}^D x_i^2}{D}}) - \exp \frac{\sum_{i=1}^D \cos(2\pi x_i)}{D} + 20 + e$	$[-5.21, 5.21]$	0
Schwefel's Problem2.22	$f_8(x) = \sum_{i=1}^D x_i + \prod_{i=1}^D x_j $	$[-10, 10]$	0
Quatic	$f_9(x) = \sum_{i=1}^D ix_i^4 + \text{random}[0, 1)$	$[-1.28, 1.28]$	0

表 2 不同测试函数寻优精度对比

函数	评估指标	PSO	GWO	GAPSO	PSOd	PSO_GWO
f_1	mean	4.59e-01	3.60e-63	1.92e-52	3.34e+03	0
	std	2.01e-01	6.77e-63	8.57e-52	1.30e+03	0
	rank	4	2	3	5	1
f_2	mean	2.35e+00	0	0	6.26e+02	0
	std	1.53e+00	0	0	2.77e+02	0
	rank	4	1	1	5	1
f_3	mean	1.21e+03	1.72e+01	1.88e+01	3.49e+04	9.11e+00
	std	1.89e+03	6.30e-01	1.61e-01	3.81e+04	9.30e+00
	rank	4	2	3	5	1
f_4	mean	1.33e+00	6.18e-03	5.50e-04	2.39e+00	0
	std	3.61e+01	1.31e-02	2.41e-03	1.05e+00	0
	rank	5	3	2	4	1
f_5	mean	1.06e+02	8.71e-01	0	1.66e+03	0
	std	3.62e+01	1.83e+00	0	1.19e+03	0
	rank	4	3	1	5	1
f_6	mean	3.61e+02	5.71e-62	3.77e-63	1.66e+03	0
	std	1.82e+03	2.29e-61	1.68e-62	1.19e+03	0
	rank	4	3	2	5	1
f_7	mean	7.81e-02	5.51e-15	8.88e-16	2.11e+00	8.88e-16
	std	4.15e-02	1.66e-15	0	4.04e+00	0
	rank	4	2	1	5	1
f_8	mean	3.54e+00	5.16e-37	2.75e-29	4.54e-01	0
	std	3.69e+00	6.22e-37	8.56e-29	2.81e-01	0
	rank	5	2	3	4	1
f_9	mean	7.83e-01	8.65e-04	6.75e-3	4.34e-01	8.35e-05
	std	4.15e-02	6.36e-04	4.87e-3	1.80e+00	6.04e-05
	rank	5	2	3	4	1
排名		39	20	19	42	9
总体排名		4	3	2	5	1

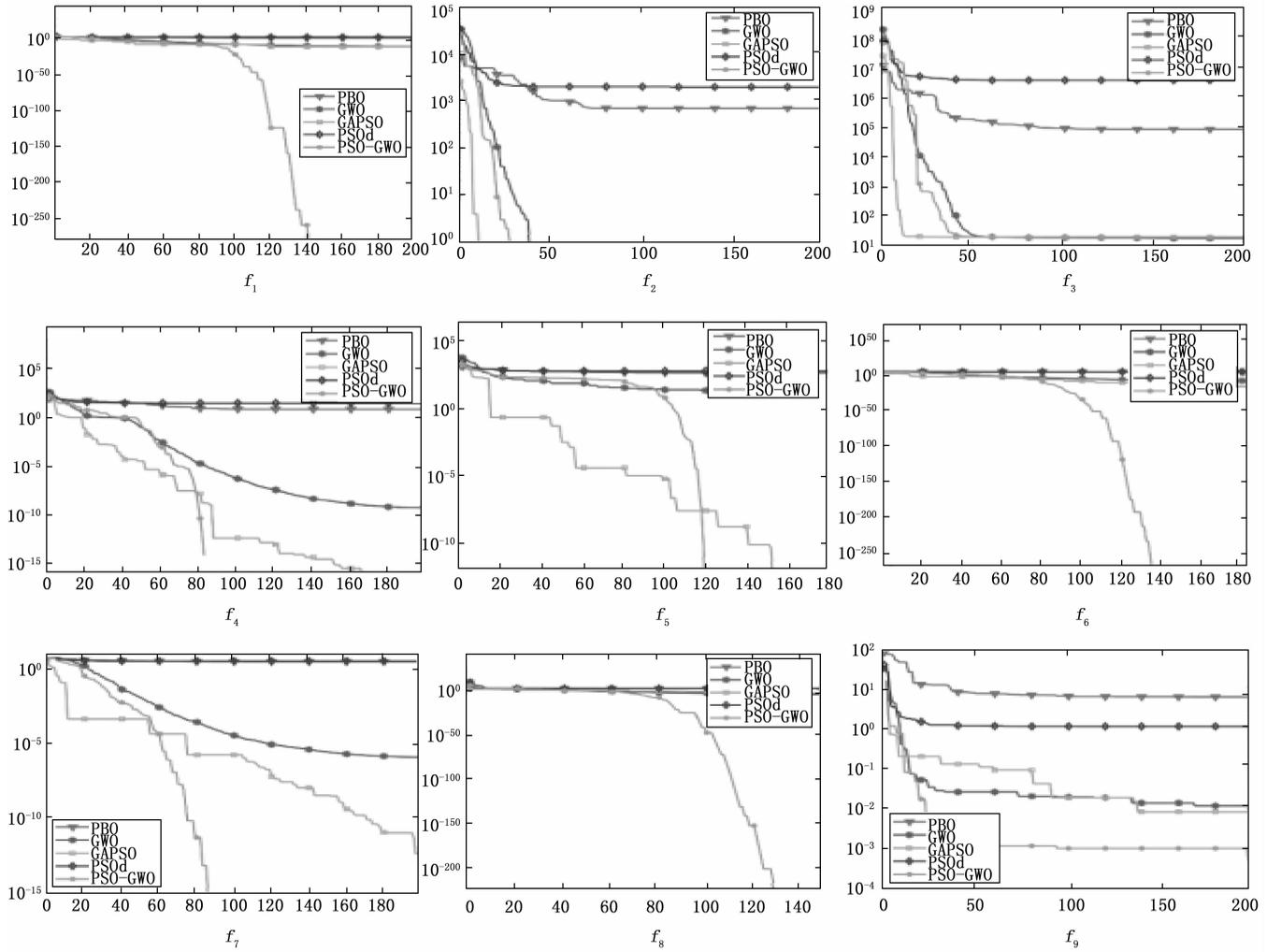


图 2 各种算法在不同函数的寻优曲线

3.2 算法收敛性能对比分析

图 2 为不同算法对不同函数寻优时的收敛曲线, 可以看出对函数 f_1 、 f_4 、 f_5 、 f_6 、 f_7 、 f_8 进行寻优时, PSO-GWO 算法在前期的收敛速度较慢, 在后期明显加快, 其中在对 f_1 函数的寻优过程中, PSO-GWO 算法在迭代 90 次时, 达到其他 4 种算法迭代 200 次的寻优精度。在对 f_4 进行寻优时, 其前期的收敛速度略差于 GAPSO, 但其在 80 次迭代时, 可达到其余 4 种算法在 200 次的寻优精度。在对 f_5 函数的寻优过程中, PSO-GWO 算法在迭代 120 次时, 达到其他 4 种算法迭代 200 次的寻优精度。在对 f_6 函数的寻优过程中, PSO-GWO 算法在迭代 80 次时, 达到其他 4 种算法迭代 200 次的寻优精度。在对 f_7 函数的寻优过程中, PSO-GWO 算法在迭代 83 次时, 达到其他 4 种算法迭代 200 次的寻优精度。在对 f_8 函数的寻优过程中, PSO-GWO 算法在迭代 70 次时, 达到其他 4 种算法迭代 200 次的寻优精度。在对 f_9 函数的寻优过程中, PSO-GWO 算法在迭代初期开始寻优速度就是优于其他 4 种算法的, 其迭代 30 次时, 达到其他 4 种算法迭代 200 次的寻优精度。

但在对 f_2 、 f_3 进行寻优时, PSO-GWO 的寻优速度虽优于 PSO、GWO、PSOd 三种算法但略差于 GAPSO。综上所述, PSO-GWO 除了在对函数 f_2 、 f_3 进行寻优时的速度略差于 GAPSO, 在对其他函数上进行寻优时, 寻优速度均是优于其他算法的。

4 结束语

本文提出了一种结合灰狼算法的粒子群优化算法 (PSO-GWO), 主要思想是对粒子群最差粒子进行进化, 提高粒子群的多样性以及对最优粒子进行扰动使粒子群跳出局部最优, 此外还结合灰狼算法的精英群体包围式搜索策略增强粒子群算法的全局搜索能力, 最后将改进完的粒子群算法 (PSO-GWO) 与标准粒子群算法 (PSO)、灰狼算法 (GWO)、遗传粒子群算法 (GAPSO)、改进位置更新公式的粒子群算法 (PSOd) 在 9 个标准测试函数上进行寻优精度和收敛性能的对。在寻优精度上, PSO-GWO 在对 9 个函数进行寻优时, 对其中 6 个函数均找到了全局最优值 0, 对其他 3 个函数进行寻优时, 即使未找到理论最优值 0, 但比较其他 4 种算法依旧提高了 1~16 个数量级。在收

