

一种协调勘探和开采能力的粒子群算法

陶新民¹, 徐 晶², 杨立标¹, 刘 玉¹

(1. 哈尔滨工程大学 信息与通信工程学院, 黑龙江 哈尔滨 150001; 2. 黑龙江科技学院 数力系, 黑龙江 哈尔滨 150027)

摘要: 提出一种新的协调勘探和开采能力的粒子群优化算法. 该算法将种群分为随机子群和进化子群, 随机子群增加了算法全局解空间的勘探能力, 在运行过程中通过随机子群进化信息生成解优胜区域指导进化粒子向着最优解子空间逼近. 为了提高算法收敛速度, 算法只在进化子群进入收敛阶段时才对其进行指导, 以防止增加种群多样性导致算法开采能力下降的问题. 将此算法与其他改进粒子群算法进行比较, 实验结果表明, 该算法有较好的全局收敛性, 不仅能有效地克服其他算法易陷入局部极小值的缺点, 而且算法收敛速度和稳定性都有显著提高.

关键词: 粒子群算法; 勘探和开采; 随机子群; 优胜区域

中图分类号: TP183 **文献标识码:** A

Particle-swarm algorithm coordinating the exploration and the exploitation

TAO Xin-min¹, XU Jing², YANG Li-biao¹, LIU Yu¹

(1. College of Information and Communication Engineering, Harbin Engineering University, Harbin Heilongjiang 150001, China;

2. Department of Mathematics and Mechanics, Heilongjiang Institute of Science and Technology, Harbin Heilongjiang 150027, China)

Abstract: A novel particle-swarm optimization(PSO) algorithm which coordinates the exploration ability and the exploitation ability(EEPSO) is presented. This algorithm divides the population of the swarm into the evolutionary sub-swarm and the randomized sub-swarm. During the evolution, the randomized sub-swarm reinforces the global space-exploration ability of the PSO algorithm, and uses the multi-species evolution information to generate the best-result-value space, guiding the particles of the evolutionary sub-swarm to approach this space. In order to improve the convergence rate, the guidance will be effective only when the evolutionary sub-swarm particles are in the convergence status. This limits the diversity of the population swarm, preventing the reduction in exploitation ability. The comparison experiments have been made between the proposed approach with the dissipative PSO and other cooperative particle swarm algorithm. The experimental results show that the proposed method not only effectively solves the premature convergence problem, but also significantly speeds up the convergence and improves the stability.

Key words: particle swarm algorithm; exploration and exploitation; randomized sub-swarm; the best result value space

1 引言(Introduction)

粒子群算法是由Kennedy和Eberhart等人于1995年提出的一种种群搜索的自适应进化计算技术^[1~3], 目前, PSO算法作为一个新的随机搜索算法, 仍存在早熟和收敛速度慢这两个难题. 为了避免早熟收敛, 一些研究者提出了通过控制种群多样性来提高算法性能的方法. 一方面, 通过解决粒子间的冲突和聚集^[4~6], 分析粒子和种群最优位置的距离, 增加环境检测和响应、种群随机多代初始化^[7,8]等思想, 给出不同策略来增强种群多样性, 使算法不至于过早陷入局部最优解, 另一方面, 通过引入遗传算法的变异操作来增强种群多样性^[9,10](dissipative

PSO, 简称DPSO), 使算法能够有效地进行全局搜索, 但是由于变异率不易控制, 过大的变异率在增加种群多样性的同时也将导致群体混乱, 使种群不能精确的局部搜索, 延缓了算法的收敛速度. 此外, 文献[11,12]利用两分群进化协同指导PSO算法(MCP SO), 帮助其逃离局部最优解. 该算法通过增强种群多样性加强了PSO算法全局解的搜索性能, 然而因分群迭代使算法计算量大大增加, 同时由于算法的进化过程破坏了PSO原有算法的粒子群结构, 使得算法的收敛速度大大降低. 因此如何在增强种群多样性的同时保持收敛速度值得我们进一步研究.

收稿日期: 2008-12-08, 收修改稿日期: 2009-05-08.

基金项目: 中国博士后科学基金资助项目(20090450119); 中国博士点新教师基金资助项目(20092304120017); 黑龙江省博士后科学基金资助项目(LBH-Z08227).

鉴于此, 本文提出一种新的协调勘探和开采能力的粒子群优化算法(EEPSO). 该算法将种群分为随机子群和进化子群, 利用随机粒子来增加算法的勘探能力, 以恢复随着迭代次数的增加而失去的种群内个体的多样性. 在兼顾勘探能力的同时为了提高算法的开采能力, 即增强精确的局部空间搜索能力, 加快收敛速度, 在进化子群进入收敛阶段时通过利用勘探得到的最优解区域信息对进化子群进行指导, 以缩小粒子的搜索区域, 加快算法的收敛速度. 利用试验对不同实际数据进行测试均显示出新算法的优良性能.

2 算法分析(Algorithm analysis)

2.1 勘探和开采(Exploration and exploitation)

任何一种进化算法都存在勘探和开采两类不同的操作, 如果这两类操作协调不好很容易使算法陷入局部最优解或降低其收敛性能. 为了防止PSO算法陷入局部最优解, 研究者通过改善种群多样性来提高算法的全局解搜索能力, 其中文献[10]等通过增加变异操作来帮助算法逃出局部最优解, 然而在算法的后期, 变异操作使粒子群不能进行精确的局部搜索, 无法保证算法的收敛性能. 文献[12]等将通过两分群的思想增强算法全局解空间的勘探能力. 由于算法对PSO进化过程全过程地指导, 打乱了原有的粒子群结构, 在算法的后期同样不能进行精确的局部搜索, 延缓了算法的收敛速度.

由于好的开采能力能保证算法进行局部精确搜索, 提高算法的收敛速度, 而提高勘探能力是为了防止算法陷入局部最优解. 因此, 在求解复杂问题时, 如何在勘探和开采之间进行有效的权衡与协调是PSO算法能否获得高性能的关键. 为此, 本文提出一种具有协调勘探和开采能力的粒子群优化算法, 不仅能保证算法的全局解空间勘探性能, 防止算法陷入局部最优, 同时在算法后期增强算法的开采能力, 使粒子群进行精确的局部搜索, 保证算法的收敛速度. 其基本思想是: 将粒子群分为2个子群: 随机子群和进化子群. 随机子群调节算法的勘探能力, 探索问题解空间以寻找那些可能最优的区域, 防止算法陷入局部最优解, 引导进化种群朝着问题解空间中可能最优的区域进行搜索; 进化子群调节算法的开采能力, 保证算法进行局部精确搜索, 提高算法的收敛速度.

2.2 随机和进化子群协调机制 (Coordination between random and evolution sub-swarms)

随机子群负责全局最优解空间的勘探, 引导进化子群向着最优解空间搜索. 为了防止增加种群多样性的同时导致进化群体原有保持的进化结构混乱,

进而使这个算法不能进行精确的局部解搜索, 最终导致算法丧失收敛性能. 本文算法通过保留随机子群和进化子群进化过程中产生的最优个体, 这里称为超级个体, 构造超级个体集合, 得到优胜区域. 当进化子群进行收敛状态时, 即第 t 代进化子群的群体适应度方差 $^{[9]}\sigma^2(t) < \varepsilon$ 时, ε 为种群进化能力阈值, 此时对超级个体优胜区域进行更新, 得到新的超级个体优胜区域, 并在新的优胜区域内生成初始粒子群, 作为新的进化子群的初始粒子重新进行迭代. 这样既保证了进化子群的粒子群结构, 同时通过引入优胜区域使得算法能够进行精确的局部搜索, 保证了算法的收敛速度, 提高了解的精度.

2.3 超级个体集合和优胜区域生成机制(Super particles and best value space generation)

在构造超级个体集合时, 为了协调算法的勘探和开采能力, 充分利用两个子群的所有有效信息, 防止种群进入收敛状态时而影响超级个体多样性, 超级个体集合由进化子群和随机子群中的最优个体组成. 当两个子群向其自身的超级个体集合提供超级个体时, 超级个体容量已达到设定值, 则用新进入的超级个体代替原有的适应度最低的超级个体, 然后重新对超级个体的适应度进行排序. 最后将两个子群的超级个体集合合并形成种群的超级个体集合.

超级个体集合所在区域表示为 $\langle I_j, L_j, U_j \rangle$, $j = 1, 2, \dots, n$, n 表示自变量的个数, 称为超级个体优胜区域. 其中参数

$$I_j = [l_j, \mu_j] = \{x_j | l_j \leq x_j \leq \mu_j, x_j \in \mathbb{R}\}$$

表示第 j 个自变量的优胜区间, L_j 和 U_j 分别表示该自变量优胜区间的下限和上限对应的适应度. 若进化子群进入收敛状态, 此时对超级个体优胜区域进行更新, 更新的规则为: 假设超级个体集合中第 i 个超级个体作用于第 j 个自变量的下限, 第 i' 个超级个体作用于第 j 个自变量的上限, 那么

$$l_j^{t+1} = \begin{cases} x_{i,j}^t, & x_{i,j}^t \leq l_j^t \text{ 或 } obj(x_i^t) < L_j^t, \\ l_j^t, & \text{其他.} \end{cases} \quad (1)$$

$$L_j^{t+1} = \begin{cases} obj(x_i^t), & x_{i,j}^t \leq l_j^t \text{ 或 } obj(x_i^t) < L_j^t, \\ L_j^t, & \text{其他.} \end{cases} \quad (2)$$

$$\mu_j^{t+1} = \begin{cases} x_{i',j}^t, & x_{i',j}^t \geq \mu_j^t \text{ 或 } obj(x_{i'}^t) < U_j^t, \\ \mu_j^t, & \text{其他.} \end{cases} \quad (3)$$

$$U_j^{t+1} = \begin{cases} obj(x_{i'}^t), & x_{i',j}^t \geq \mu_j^t \text{ 或 } obj(x_{i'}^t) < U_j^t, \\ U_j^t, & \text{其他.} \end{cases} \quad (4)$$

式中: l_j^t 和 μ_j^t 表示超级个体集合中第 t 代第 j 个

自变量的下限和上限, L_j^t 和 U_j^t 分别表示其适应度, $x_{i,j}^t$ 表示第 t 代第 i 个个体的第 j 个自变量, obj 表示为适应度函数。

2.4 EEP SO算法流程 (EEP SO algorithm design)

设 N 代表种群总的粒子个数, EP_{Li} 表示进化子群中该粒子目前为止经过的最好位置, EP_g 表示进化子群中所有粒子到目前為止经历过的最好位置; RP_j 表示随机子群中的粒子。协调勘探和开采的粒子群算法流程可表述如下:

1) 初始化粒子群, 并将其分成2个子群: $\gamma * N$ 随机子群和 $(1 - \gamma) * N$ 进化子群, γ 为参数;

2) 在超级个体优胜区域 $[L, U]$ 内, 随机生成粒子 $RP_j, j \in (1, 2, \dots, S)$;

3) 对进化子群中的粒子 $EP_i, i \in (1, 2, \dots, (1 - \gamma) * N)$ 进行进化;

4) 如果 $f(EP_i) < f(EP_{Li})$ 则

$$EP_{Li} = EP_i;$$

5) 如果 $f(EP_{Li}) < f(EP_g)$ 则

$$EP_g = EP_{Li};$$

6) 利用粒子更新公式更新粒子 EP_i ;

7) 更新超级个体集合, 对随机子群和进化子群分别进行适应度比较, 并将结果合并形成新的超级个体集合;

8) 判断进化子群的进化状态, 如果 $\sigma^2(t) < \varepsilon$, 根据公式(1)~(4)更新超级个体优胜区域;

9) 在超级个体优胜区域 $[L, U]$ 内, 随机生成初始粒子群并根据优胜区域调整惯性参数 w , 重新进行进化。为了保证全局收敛, 将最优超级个体代替本次进化子群中的最差粒子进入下一次迭代;

10) 符合算法停止条件, 输出超级个体集合中的最优个体, 作为问题的最优解。

3 试验分析(Experimental analysis)

3.1 Benchmark函数(Benchmark functions)

为了分析本文提出的新算法的全局搜索性能、收敛速度和算法的稳定性, 选择进化算法经常使用的3个Benchmark函数问题进行数值试验, 单模态Rosenbrock和多模态Griewank, Rastrigin函数, 下列公式给出了函数定义、取值范围和全局最优解。

Rosenbrock函数

$$f_1 = \sum_{i=1}^n (100(x_{i+1} - x_i^2) + (x_i - 1)^2), \quad (5)$$

取值范围为 $[-30, 30]$, 维度为10, 最小值和位置为 $(1, \dots, 1)$ 。

多模态Griewank函数

$$f_2 = \frac{1}{4000} \sum_{i=1}^n (x_i^2) - \prod_{i=1}^n (\cos(x_i/\sqrt{i})) + 1, \quad (6)$$

取值范围为 $[-30, 30]$, 维度为20, 最小值和位置为 $(0, \dots, 0)$ 。

多模态Rastrigin函数

$$f_3 = \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i) + 10), \quad (7)$$

取值范围为 $[-30, 30]$, 维度为10, 最小值和位置为 $(0, \dots, 0)$ 。

3.2 单模态函数优化对比试验(Performance comparison of single model function)

设置本文算法和文献[12]算法的子群个数为2, 粒子个数为50, 粒子的维度为10, 迭代次数为20000代, 群体适应度方差的阈值为0.0001, 本文算法中 γ 取0.45。为了排除算法内部随机操作对性能的影响, 以1000次试验的统计结果进行分析。利用有代表性的文献[10]基于变异操作的DPSO、文献[12]的基于分群的MCP SO算法和本文的EEP SO3种算法对Rosenbrock问题求解, 结果如图1所示。

从图1可以看出, 在算法的初期阶段, 由于Rosenbrock函数山谷仅仅给算法提供了很少的信息, 使算法不能有效地辨别搜索方向, 图1中所有算法都具有相对缓慢下降的阶段, 用来寻找搜索方向。试验中发现对于Rosenbrock函数, 本文算法在1000次试验中有20%的机会逃出0.0104局部最优解。这是由于随机子群的引入, 使得算法的解空间勘探能力提高, 算法朝着最优的解空间进化, 较好地逃出局部最优解。

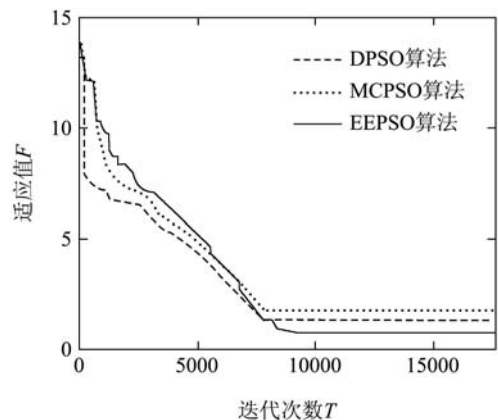


图1 10维Rosenbrock收敛性能对比图

Fig. 1 Comparison of performance of convergence on 10-dimension Rosenbrock function

表1中3种算法对单模态Rosenbrock求解问题的均值、方差的数据显示, 本文算法的均值和方差数据均优于其他算法, 显示了新算法的稳定性和健壮性。

表 1 Rosenbrock(10)函数的优化性能数据表

Table 1 Performance of optimization on Rosenbrock

优化算法	平均值	方差	最大值	最小值
DPSO算法	1.2046	0.9264	2.4889	0.0164
MCPSO算法	0.8079	0.6068	1.6038	0.0104
EEPSO算法	0.5378	0.3924	1.0380	0.0053

3.3 多模态函数优化对比试验(Performance comparison of multi model function)

对2个多模态问题优化结果如图2, 3所示, 从中可以看出本文算法优于其他两种算法, 具有良好的全局最优解搜索能力和较快的收敛速度. 这是由于本文算法协调了勘探和开采能力, 增加了随机子群的指导, 扩大了解空间的勘探搜索范围, 使得搜索方向向着最优解子空间逼近, 并且对最优解区域进行更新, 使得算法具有较快的收敛速度.

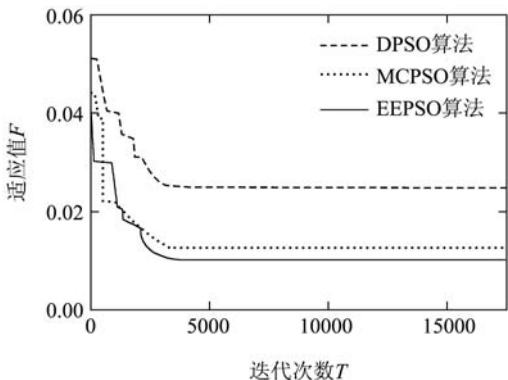


图 2 20维Griewank函数收敛性能对比图

Fig. 2 Comparison of performance of convergence on 20-dimension Griewank function

试验中可以看出, 由于随机和进化种群通过优胜区域的合理协调, 算法收敛速度也有所提高.

表 2 Griewank函数的优化性能数据表

Table 2 Performance of optimization on Griewank

优化算法	平均值	方差	最大值	最小值
DPSO算法	0.1094	0.0524	0.1574	0.0468
MCPSO算法	0.0744	0.0261	0.1058	0.0492
EEPSO算法	0.0652	0.0263	0.1033	0.0443

多模态Rastrigin函数是一个经典的用来测试全局搜索性能的函数. 图3显示DPSO算法和MCPSO算法在搜索后期容易陷入局部最优解, 而本文算法的随机子群多次指导进化子群逃出局部最优解. 表2, 3中本文算法的均值和方差数据均优于其他算法, 显示了新算法的稳定性和健壮性. 结合上面的试验可以看出, 由于本文算法通过将粒子群分为2个子群,

增加了算法的种群多样性, 使得算法全局寻优能力大大增强, 克服了原有算法陷入局部最优解的不足; 同时通过优胜区域的引入使得算法勘探与开采性能得以协调; 在保证算法全局解性能的前提下, 提高了了解的精度及算法收敛速度.

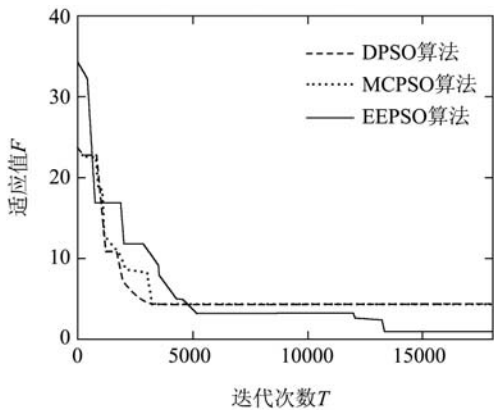


图 3 10维Rastrigin函数收敛性能对比图

Fig. 3 Comparison of performance of convergence on 10-dimension Rastrigin function

表 3 Rastrigin函数的优化性能数据表

Table 3 Performance of optimization on Rastrigin

优化算法	平均值	方差	最大值	最小值
DPSO算法	4.4536	1.7796	7.2016	0.9950
MCPSO算法	3.2928	2.6058	7.2016	0.9950
EEPSO算法	1.4472	0.6841	2.9849	0.9950

4 结论 (Conclusion)

本文提出一种新的协调勘探和开采能力的粒子群优化算法. 结合试验得到如下结论:

1) 引入控制参数将粒子群分为随机子群和进化子群, 调节算法的勘探能力和开采能力, 使其具有更大的灵活性. 并且参数容易确定, 通过对参数进行调整, 可以使算法在勘探能力, 即全局解搜索性能及收敛速度上达到最优平衡.

2) 利用种群进化信息生成解的优胜区域, 找出更有效的最优解子空间, 使得进化子群的粒子向着最优解空间逼近, 在进化过程中始终保持最优解空间勘探的能力, 同时保持了算法的收敛速度.

3) 利用不同特征的数据分析了算法的全局解搜索性能以及收敛速度, 同时与DPSO算法和MCPSO算法进行比较, 试验结果表明本文算法有效提高了全局解搜索能力, 并且算法很稳定.

需要指出的是, 本文算法中种群进化能力阈值对全局解空间搜索能力的影响以及不同的种群大小对

实验结果的影响尚没有进行讨论,这也是本课题下一步研究的重点.

参考文献(References):

- [1] VAN DEN BERGH F, ENGELBRECHT A P. A new locally convergent particle swarm optimization[C] // *Proceeding of IEEE International Conference on System, Man and Cybernetics*. Hammamet: IEEE, 2002, 8: 129 – 134.
- [2] CLERC M, KENNEDY J. The particle swarm explosion, stability, and convergence in a multidimensional complex space[J]. *IEEE Transactions on Evolutionary Computation*, 2002, 6(2): 58 – 73.
- [3] MENDES R, KENNEDY J, NEVES J. The fully informed particle swarm: simpler, maybe better[J]. *IEEE Transactions on Evolutionary Computation*, 2004, 8(6): 204 – 210.
- [4] 孟红记, 郑鹏. 基于混沌序列的粒子群优化算法[J]. 控制与决策, 2006, 21(3): 263 – 266.
(MENG Hongji, ZHENG Peng. Particle swarm optimization algorithm based on chaotic series[J]. *Control and Decision*, 2006, 21(3): 263 – 266.)
- [5] 陈国初, 俞金寿. 两群微粒群优化算法及其应用[J]. 控制理论与应用, 2007, 24(2): 294 – 298.
(CHEN Guochu, YU Jinshou. Two sub-swarm particle swarm optimization algorithm and its application[J]. *Control Theory & Applications*, 2007, 24(2): 294 – 298.)
- [6] PARSOPOULOS KE, VRAHATIS MN. On the computation of all global minimizers through particle swarm optimization[J]. *IEEE Transactions on Evolutionary Computation*, 2004, 8(3): 211 – 224.
- [7] TRELEA IC. The particle swarm optimization algorithm: convergence analysis and parameter selection[J]. *Information Processing Letters*, 2003, 85(9): 317 – 325.
- [8] 彭宇, 彭喜元, 刘兆庆. 微粒群算法参数效能的统计分析[J]. 电子学报, 2004, 32(3): 209 – 213.
(PENG Yu, PENG Xiyuan, LIU Zhaoqing. Statistic analysis on parameter efficiency of particle swarm optimization[J]. *Acta Electronica Sinica*, 2004, 32(3): 209 – 213.)
- [9] 吕振肃, 侯志荣. 自适应变异的粒子群优化算法[J]. 电子学报, 2004, 32(3): 416 – 420.
(LÜ Zhensu, HOU Zhirong. Particle swarm optimization with adaptive mutation[J]. *Acta Electronica Sinica*, 2004, 32(3): 416 – 420.)
- [10] 赫然, 王永吉. 一种改进的自适应逃逸微粒群算法及实验分析[J]. 软件学报, 2005, 16(12): 2036 – 2044.
(HE Ran, WANG Yongji. An improved particle swarm optimization based on self-adaptive escape velocity[J]. *Journal of Software*, 2005, 16(12): 2036 – 2044.)
- [11] BERGH F, ENGELBRECHT A P. A cooperative approach to particle swarm optimization[J]. *IEEE Transactions on Evolutionary Computation*, 2004, 8(3): 1 – 15.
- [12] 王俊年, 申群太. 基于多种群协同进化微粒群算法的径向基神经网络设计[J]. 控制理论与应用, 2006, 32(2): 251 – 255.
(WANG Junnian, SHEN Quntai. Evolutionary design of RBF neural network based on multi-species cooperative particle swarm optimizer[J]. *Control Theory & Applications*, 2006, 32(2): 251 – 255.)

作者简介:

陶新民 (1973—), 男, 博士, 副教授, 主要研究软计算、信号检测等,

E-mail: taixinmin@hrbeu.edu.cn;

徐晶 (1974—), 女, 硕士, 副教授, 主要研究方向为进化计算;

杨立标 (1981—), 男, 硕士研究生, 主要研究方向为进化计算;

刘玉 (1985—), 男, 硕士研究生, 主要研究方向为模式识别.