

# 一种改进的粒子群多目标优化算法研究

刘慧慧

(南京邮电大学 自动化学院, 江苏 南京 210046)

**摘要:**为了解决多目标优化过程中各个解之间存在的资源争夺、冲突,算法由于趋同性而带来的早熟无法收敛等缺点,文中提出了一种多子种群协同优化粒子群算法。算法分别采用不同的种群优化不同的目标,并且在算法中引入外部档案和精英学习策略,使得算法能够得到更多的外部档案的解供选择,精英学习策略是为了使算法的分布性和收敛性更好。最后将算法应用到多目标测试函数中,通过实验验证了改进后的算法的收敛性和分布性都比经典多目标算法 NSGA-II 要好。

**关键词:**多目标优化;粒子群算法;多子种群;外部档案

中图分类号:TP31

文献标识码:A

文章编号:1673-629X(2015)01-0087-04

doi:10.3969/j.issn.1673-629X.2015.01.020

## Research on an Improved Multi-objective Optimization Algorithm of Particle Swarm

LIU Hui-hui

(School of Automation, Nanjing University of Posts and Telecommunications,  
Nanjing 210046, China)

**Abstract:** To solve the problem that resource contention and conflict between the various solutions in multi-objective optimization processing, and can't be convergence duo to the precocious brought by convergence, introduce a multi-sub-population co-evolution mechanism to overcome these shortcomings. The algorithm has adopted different populations to optimize different targets. Meanwhile, it introduces an external archive and elite learning strategies, in this way it can obtain more solutions of external archive to choose. Elite learning strategies makes the algorithm has a better distribution and convergence. Finally, the algorithm is applied into the multi-objective test function, the experimental results show that the improved algorithm has a better convergence and distribution than NSGA-II.

**Key words:** multi-objective optimization; particle swarm algorithm; multi-sub-population; external archive

## 0 引言

粒子群算法已经成功地应用于单目标优化,而在实际的工程应用中很多都要解决多目标优化的问题,所以越来越多的研究者已经将其扩展应用到多目标优化。多目标优化问题由于其各个解之间的冲突,不存在唯一的解,而是一组 Pareto 最优解集,所以如何选择局部最优和全局最优很重要,而且由于其不可微、不连续、非线性等特点,传统的数学方法已经很难解决此类问题。此时,智能计算方法在该问题中表现出很大的优势。传统的智能算法也只是将多目标进行加权将其转化为单目标,而该算法对加权系数的要求较高。

文献[1]采用了 NSGA-II (Nondominated Sorting

Genetic Algorithm II) 来解决多目标问题并计算其收敛性和分布性。文献[2]采用 PSO (Particle Swarm Optimization) 动态邻域策略更新粒子,需要指定共享参数的问题。文献[3]采用外部档案存储非劣解。文献[4]提到了采用混合算法来解决多目标优化问题。文献[5]提出了一种协同进化机制来解决多目标问题以及传统进化算法中适应度的分配问题,每个种群代表一个目标函数。文献[6]还采用了小生境技术与部分变异相融合来提高解的多样性。文献[3,6-9]中都采用了遗传算法中的变异操作和粒子群算法的融合,文献[8]还引进了拥挤距离排序与强支配排序来构造外部档案存储非劣解,提高解的多样性,文献[9]还引入

了交叉操作。

文中在已有论文的基础上,针对多目标优化算法所存在的不足,引入多子种群协同进化机制,解决多目标优化中各个解之间的冲突问题。各个子种群之间相互独立又个体协作,从而使粒子共同朝着最优方向飞行。粒子群主要通过一个外部档案进行信息的共享。同时,为了提高算法的多样性,采取精英选择策略更新外部档案中的粒子。随着非劣解的增加,外部档案的容量越来越大,会影响算法的效率,所以外部档案采取固定容量,当外部档案已满,采用拥挤距离机制对外部档案进行削减,提高算法的效率。文中主要通过对粒子群算法进行改进解决多目标优化问题中解的收敛性和多样性问题。

## 1 多目标优化数学模型

模型如下:

$$\begin{cases} y = F(x) = \min (f_1(x), f_2(x), \dots, f_m(x))^T \\ \text{st: } g_i(x) \leq 0, i = 1, 2, \dots, p \\ h_j(x) = 0, j = 1, 2, \dots, q \end{cases}$$

其中,  $x = (x_1, x_2, \dots, x_n)^T$ ,  $x$  为  $n$  维变量,  $m$  为目标个数,  $g(x)$ ,  $h(x)$  为约束条件。

## 2 粒子群优化算法

粒子群算法 (PSO) 模拟鸟类的觅食行为,通过鸟类之间的合作使群体达到最优<sup>[10]</sup>。鸟类通过自身的学习及向其邻域内的粒子学习来搜索最优解。

### 2.1 PSO 算法的数学模型

设算法的搜索空间为  $D$  维,粒子的个数为  $n$ 。速度和位置公式如下:

$$\begin{aligned} v_{id}^{t+1} &= \omega v_{id}^t + c_1 r_1 (v_{id}^t - pbest) + c_2 r_2 (v_{id}^t - gbest) \\ x_{id}^{t+1} &= x_{id}^t + v_{id}^{t+1} \end{aligned}$$

其中,  $v_{id}^t$  为第  $t$  代第  $i$  个粒子的当前速度;  $pbest$  表示第  $t$  代粒子的个体最优粒子;  $gbest$  为叠加至  $t$  代所得到的全局最优粒子;  $v_{id}$  表示第  $i$  个粒子的速度;  $\omega$  表示维持当前速度的惯性权重系数;  $c_1$  和  $c_2$  表示加速因子;  $r_1$  和  $r_2$  表示  $[0, 1]$  之间的随机数。

### 2.2 多子种群协同进化

由于多目标优化要使得所有的目标同时达到最大值,各个目标之间存在着冲突问题,所有文中采用多个子种群协同进化,各个子种群之间通过信息的共享,使得算法能够更快更方便地找到最优解。

在多子种群协同进化过程中,有几个目标就采用几个种群对各个目标进行优化,若有  $n$  个目标,每个目标记为  $S_i$ ,  $i = 1, 2, \dots, n$ , 每个粒子的飞行方向不仅仅由粒子的个体最优和全局最优决定,还由其他种群粒

子的飞行速度决定,粒子间的信息共享主要通过外部档案来实现。

多子种群协调优化结构图如图 1 所示。

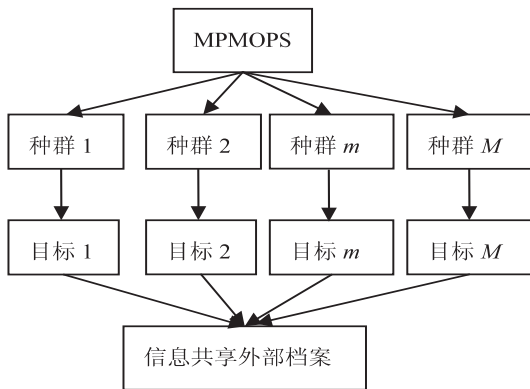


图 1 多子种群协同优化结构图

图 1 中显示出  $M$  个种群优化  $M$  个目标的框架图。在每一次迭代中,每个种群的粒子采用传统的粒子群算法优化每个目标。在进行粒子的选择过程中,第  $m$  个种群的粒子的适应度值根据第  $m$  个目标函数来确定,  $1 \leq m \leq M$ 。采用多子种群来优化多目标,每个种群选择的最好的粒子不会与其他种群的粒子存在冲突,然而粒子之间又有协作,共同搜索不同的空间,从而使算法趋近 Pareto 前沿。

为了实现各个种群之间信息的共享,因此需要设计一个外部的档案,这样使得算法更加快速有效地收敛至 Pareto 前沿。

### 2.3 外部档案

对于多目标优化问题,无法像单目标一样得到一个最优解,多目标问题所得到的解对于各个目标之间存在冲突,所以对得到的非劣解集采用一个外部档案进行存储。

$$\begin{aligned} v_{id}^{t+1} &= \omega v_{id}^t + c_1 r_1 (v_{id}^t - pbest) + c_2 r_2 (v_{id}^t - gbest) + \\ &\quad c_3 (v_{id}^t - A_{id}) \\ x_{id}^{t+1} &= x_{id}^t + v_{id}^{t+1} \end{aligned}$$

其中,  $A_{id}$  表示外部档案中的粒子。在该算法中,外部档案不仅实现了对非劣解的存储,同时也实现了信息的共享,在这里  $A_{id}$  采用外部档案中的均值,这样使粒子能够对所有的目标更容易达到最优值。

随着迭代次数的增加,采用得到的新解不断地更新外部档案。然后随着外部档案中的解不断的增大,计算量也将不断增大。为了克服这一缺点,当外部档案达到最大值时,采用拥挤距离机制对外部档案进行削减。

### 2.4 精英选择策略更新外部档案

精英学习策略 Elitist\_Learning\_Strategy (ELS) 最开始在文献[10]中自适应粒子群中引入来使得粒子跳出局部最优。文中采用 ELS 来更新外部档案中的粒

子,因为它们全是全局最优粒子。对于档案 A 中的粒子采用如下办法:

$$E_{id} = E_{id} + (X_{\max, d} - X_{\min, d}) \text{Gaussian}(0, 1)$$

由于有新粒子的引入,所以会增加粒子的多样性。

### 2.5 基于拥挤距离削减外部档案

随着外部档案中非劣解的个数增多,外部档案的容量增大,计算量也增大。为了提高计算效率,对于外部档案 A,采取固定容量 NA,当容量大于 NA 时,采用基于拥挤距离对外部档案进行削减。

将得到的非劣解放入集合 R 中:

```

L = size(R)
for each solution in R
    di = 0;
end of for;
for m = 1:M //each objective
    di = max, dL = max;
    for i = 2:L-1
        di = di + (Ri+1 · Fm - Ri-1 · Fm) / (Fm, max - Fm, min);
    end of for
end of for
sort R according to the di;
select the first NA solution with large di into the archive A
end

```

### 2.6 整个算法过程

对于 M 个种群,每个种群有 N 个粒子,对这 MN 个粒子进行初始化,外部档案 A 被设置为空。对于第 m 个种群的第 i 个粒子,随机初始化其速度和位置为  $V_i^m$ , position  $X_i^m$ ,然后根据第 m 个目标计算  $X_i^m$ , pBest<sub>i</sub><sup>m</sup> 设置为  $X_i^m$ 。gBest<sup>m</sup> 由 pBest<sub>i</sub><sup>m</sup> 的最好位置决定。初始化之后,外部档案 A 也确定,算法开始迭代。

在每一次迭代过程中,第 m 个目标第 i 个粒子随机从外部档案 A 中选择一个解,然后根据公式来更新其速度和位置。所有的粒子的位置更新后,如果第 m 个目标的  $X_i^m$  比 pBest<sub>i</sub><sup>m</sup> 好,则用其替换 pBest<sub>i</sub><sup>m</sup>,如果 pBest<sub>i</sub><sup>m</sup> 在第 m 维目标中具有更小的适应度值, gBest<sup>m</sup> 也将会被替换,每一代结束后更新外部档案 A。当算法满足结束条件时结束,保存在外部档案 A 中的即是最优解。

### 2.7 算法性能的判断

算法的分布性指标 ( $\Delta$ ) 和收敛性指标 ( $\gamma$ )<sup>[11]</sup> 如下:

$$\gamma = \frac{\sqrt{\sum_{i=2}^N d_i^2}}{N}$$

$$\Delta = \frac{\sum_{m=2}^M d_m^e + \sum_{i=2}^{N-2} |d_i - \text{avg}(d)|}{\sum_{m=2}^M d_m^e + (N-1) \text{avg}(d)}$$

其中,  $d_i$  表示通过算法所得的非劣解与真实的 Pareto 解之间的距离; N 表示算法所得到的非劣解的个数; 由公式可以看出  $\gamma$  越小, 所得到的非劣解越接近真实的解, 若  $\gamma$  为 0, 表示所得到的解为真实的 Pareto 解。  $\gamma$  越小, 收敛性越好<sup>[12]</sup>;  $d_m^e$  表示 Pareto 前沿的解与所得到的解对应的第 m 个目标之间的距离; avg(d) 表示所有  $d_i$  的平均值;  $\Delta$  越小, 算法所得到的解的分布性越好<sup>[13]</sup>。

## 3 实验结果及分析

测试函数:

采用 ZDT1-6<sup>[14]</sup> 测试函数在收敛性和分布性方面对该算法与 NSGAII 进行比较。

$$\text{ZDT1} = \begin{cases} f_1(x) = x \\ f_2(x) = g(x) (1 - \sqrt{x_1/g(x)}) \end{cases}$$

其中,  $g(x) = 1 + 9(\sum_{i=2}^n x_i)/(n-1)$ ,  $x_i \in [0, 1]$ ,  $n = 30$ 。

ZDT1 所得到的 Pareto 前沿为一条凸曲线。

$$\text{ZDT2} = \begin{cases} f_1(x) = x \\ f_2(x) = g(x) (1 - (x_1/g(x))^2) \end{cases}$$

其中,  $g(x) = 1 + 9(\sum_{i=2}^n x_i)/(n-1)$ ,  $x_i \in [0, 1]$ ,  $n = 30$ 。

ZDT2 所得到的 Pareto 前沿为非凸曲线。

$$\text{ZDT3} = \begin{cases} f_1(x) = x \\ f_2(x) = g(x) (1 - \sqrt{x_1/g(x)} - \frac{x_1}{g(x)} \sin 10\pi x_1) \end{cases}$$

其中,  $g(x) = 1 + 9(\sum_{i=2}^n x_i)/(n-1)$ ,  $x_i \in [0, 1]$ ,  $n = 30$ 。

ZDT3 的 Pareto 为离散的曲线。

$$\text{ZDT4} = \begin{cases} f_1(x) = x \\ f_2(x) = g(x) (1 - \sqrt{x_1/g(x)}) \end{cases}$$

其中,  $g(x) = 1 + 10(n-1) + \sum_{i=2}^n (x_i^2 - 10 \cos 10\pi x_i)$ ,  $x_1 \in [0, 1]$ ,  $x_i \in [-5, 5]$ ,  $n = 10$ 。

ZDT4 的 Pareto 前沿比较复杂。

$$\text{ZDT6} = \begin{cases} f_1(x) = x - \exp(-4x_1) (\sin 6\pi x_1)^6 \\ f_2(x) = g(x) [1 - (x_1/g(x))^2] - 0.25 \end{cases}$$

其中,  $g(x) = 1 + 9(\sum_{i=2}^n x_i)/(n-1)$ ,  $x_i \in [0, 1]$ ,  $n = 10$ 。

整个算法所得到的 Pareto 前沿与真实的 Pareto 前沿的对比如图 2 所示。

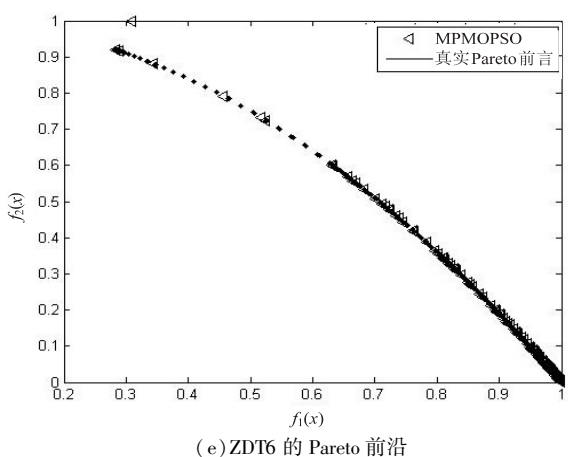
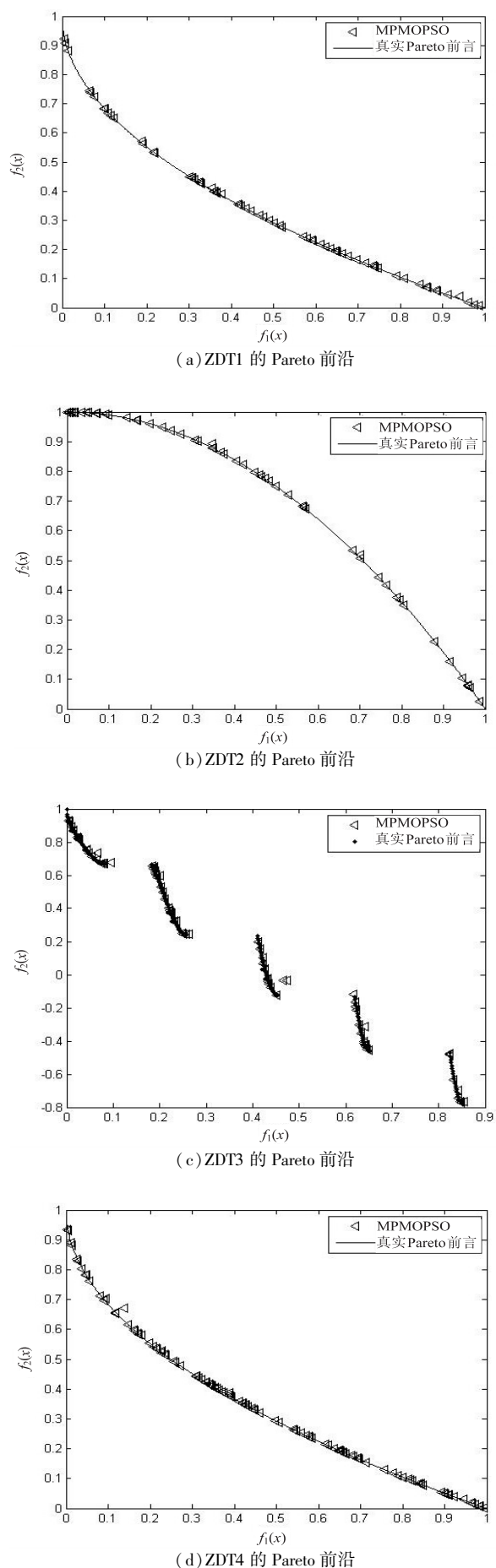


图 2 算法所得到 Pareto 前沿与真实的 Pareto 前沿的对比

通过实验以及算法的收敛性和分布性比较结果如表 1 和表 2 所示。

表 1 各算法收敛性 GD

函数	算法		
	MPMOPSO	MOPSO	NSGA_II
ZDT1	0.014 572	0.089	0.033 482
ZDT2	0.042 279	0.381	0.072 391
ZDT3	0.101 245	0.215	0.114 500
ZDT4	0.321 456	0.719	0.513 053
ZDT6	0.210 102	0.532	0.296 564

表 2 各算法的分布性 SP

函数	算法		
	MPMOPSO	MOPSO	NSGA_II
ZDT1	0.191 23	0.72	0.390 37
ZDT2	0.092 81	0.74	0.437 76
ZDT3	0.421 03	0.63	0.738 540
ZDT4	0.319 28	0.81	0.702 612
ZDT6	0.234 91	0.92	0.668 025

由上面测试函数所得的 Pareto 前沿可以看出无论是收敛性还是分布性,改进的算法所得到的结果都还是不错的。由此证明了算法的可行性。

4 结束语

文中通过在粒子群算法中引入多子种群协调优化操作,解决了多目标优化问题中多个目标相互冲突的问题,使得两个及两个以上目标的优化成为可能。同时通过使用的外部档案在存储非劣解的同时还实现了信息的共享。为了增加外部档案的多样性,采用精英学习策略增加多样性。最后通过 ZDT 系列的测试函数,验证了改进的算法在多目标优化中的多样性和收敛性。证明了该算法是可行的。

优性能,并且获得相对快的收敛速度,所以算法将改进的快速模拟退火机制引入到精英协同进化中,并减少了精英操作算子;同时 SACEA 中随机种群的加入增加了种群的多样性,更好地防止了早熟收敛。文中通过对 15 组标准测试函数的仿真,并与 MECA 算法作对比,体现了 SACEA 算法在全局寻优性能、收敛性以及收敛速率方面的更好表现。

参考文献:

[1] Potter M A, de Jong K A. A cooperative coevolutionary approach to function optimization [C]//Proc of the parallel problem solving from nature. [ s. l. ] : [ s. n. ], 1994 : 249 – 257.

[2] Danoy G, Bouvry P, Martins T. hLCGA: a hybrid competitive coevolutionary genetic algorithm [C]//Proc of the 6th international conference on hybrid intelligent systems. Auckland: IEEE Computer Society Press, 2006: 48–51.

[3] 王 磊, 刘小勇. 协同人工免疫计算模型的研究 [J]. 电子学报, 2009, 37 (8) : 1739–1745.

[4] Meng W, Han X D, Hong B R. Bee evolutionary genetic algorithm [J]. Acta Electronica Sinica, 2006, 34 (7) : 1294–1300.

[5] Mu Caihong, Jiao Licheng, Liu Yi. M-Elite coevolutionary al-

gorithm for numerical optimization [J]. Journal of Software, 2009, 20 (11) : 2925–2938.

[6] 刘 全, 王晓燕, 傅启明, 等. 双精英协同进化遗传算法 [J]. 软件学报, 2012, 23 (4) : 765–775.

[7] 周昌乐, 游 维, 丁晓君. 一种宋词自动生成的遗传算法及其机器实现 [J]. 软件学报, 2010, 21 (3) : 427–437.

[8] 应伟勤, 李元香, Shen Philip CY, 等. 演化多目标优化中的几何热力学选择 [J]. 计算机学报, 2010, 33 (4) : 755–767.

[9] 朱颢东, 钟 勇. 一种改进的模拟退火算法 [J]. 计算机技术与发展, 2009, 19 (6) : 32–35.

[10] 王庆荣, 袁占亨, 张秋余. 基于改进遗传—模拟退火算法的公交排班优化研究 [J]. 计算机应用研究, 2012, 29 (7) : 2461–2463.

[11] 高 鹰, 谢胜利. 基于模拟退火的粒子群优化算法 [J]. 计算机工程与应用, 2004, 40 (1) : 47–50.

[12] Metropolis N, Rosenbluth A W, Rosenbluth M N, et al. Equation of state calculations by fast computing machines [J]. Journal of Chemical Physics, 1953, 21 (6) : 1087–1092.

[13] Kirkpatrick S, Gelatt J C D, Vecchi M P. Optimization by simulated annealing [J]. Science, 1983, 220 (4598) : 671–680.

[14] Rudolph G. Convergence analysis of canonical genetic algorithms [J]. IEEE Transactions on Neural Networks, 1994, 5 (1) : 96–101.

(上接第 90 页)

参考文献:

[1] Qian F S A. Application of fast and elitist non-dominated sorting generic algorithm in multi-objective reactive power optimization [J]. Transactions of China Electro Technical Society, 2007, 12: 24–24.

[2] Han Min, Fan Jianchao. Particle swarm optimization using dynamic neighborhood topology for large scale optimization [C]//Proc of 2010 8th world congress on intelligent control and automation. Jinan: IEEE, 2010, 3138–3142.

[3] Coello C A C, Pulido G T, Lechuga M S. Handling multiple objectives with particle swarm optimization [J]. IEEE Trans on Evolutionary Computation, 2004, 8 (3) : 256–279.

[4] Jiao Jian, Wang Xianjia, Zhang Liubo. Applying particle swarm optimization in multiobjective optimization and hybrid optimization [C]//Proc of 2010 second international conference on computational intelligence and natural computing. [ s. l. ] : [ s. n. ], 2010.

[5] Zhang Jun, Chung H S H, Lo W L, et al. Pseudo co-evolutionary genetic algorithms for power electronic circuits optimization [J]. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, 2006, 36 (4) : 590 –

598.

[6] 李 宁, 邹 彤, 孙德宝, 等. 基于粒子群的多目标优化算法 [J]. 计算机工程与应用, 2005, 41 (23) : 43–46.

[7] 刘衍民, 牛 奔, 赵庆祯. 多目标优化问题的粒子群算法仿真研究 [J]. 计算机应用研究, 2011, 28 (2) : 458–460.

[8] 胡广浩, 毛志忠, 何大阔. 基于两阶段领导的多目标粒子群优化算法 [J]. 控制与决策, 2010, 25 (3) : 404–410.

[9] 刘衍民, 牛 奔, 赵庆祯. 基于交叉和变异的多目标粒子群算法 [J]. 计算机应用, 2011, 31 (1) : 82–84.

[10] 盛跃宾, 陈定昌, 穆 森, 等. 有等式约束优化问题的粒子群优化算法 [J]. 计算机工程与设计, 2006, 27 (13) : 2412–2413.

[11] 陈民铀, 张聪誉, 罗辞勇. 自适应进化多目标粒子群优化算法 [J]. 控制与决策, 2009, 24 (12) : 1851–1855.

[12] 陶新民, 徐 晶, 杨立标, 等. 一种改进的粒子群和 K 均值混合聚类算法 [J]. 电子与信息学报, 2010, 32 (1) : 92–97.

[13] 徐 刚, 杨玉群, 黄先玖. 一种非线性权重的自适应粒子群优化算法 [J]. 计算机工程与应用, 2010, 46 (35) : 49–51.

[14] Zitzler E, Ded K, Thiele L. Comparison of multiobjective evolutionary algorithms: empirical results [J]. Evolutionary Computation, 2002, 8 (2) : 173–195.

# 一种改进的粒子群多目标优化算法研究

作者：[刘慧慧](#)，[LIU Hui-hui](#)  
作者单位：[南京邮电大学 自动化学院, 江苏 南京, 210046](#)  
刊名：[计算机技术与发展](#)[ISTIC](#)  
英文刊名：[Computer Technology and Development](#)  
年，卷(期)：2015(1)

引用本文格式：[刘慧慧](#),[LIU Hui-hui](#) 一种改进的粒子群多目标优化算法研究[期刊论文]-[计算机技术与发展](#)  
2015(1)