

# 多目标混沌进化算法

雷德明<sup>1,2</sup>, 严新平<sup>1</sup>, 吴智铭<sup>3</sup>

(1. 武汉理工大学智能运输系统研究中心, 湖北武汉 430063;  
2. 武汉理工大学自动化学院, 湖北武汉 430070; 3. 上海交通大学自动化研究所, 上海 200030)

**摘要:** 设计了多目标混沌进化算法(MCEA), 在每一代遗传操作和外部档案调整完成之后, 该算法从外部档案中随机选择部分个体, 对这些个体的拷贝进行混沌搜索, 以产生更多非劣解. 将强度 Pareto 进化算法(SPEA)和 SPEA2 分别与基于 Logistic 映射的混沌搜索结合而产生的 MCEAs 应用于一些复杂多目标优化问题, 计算结果表明, 混沌的加入, 明显改善了多目标进化算法(MOEA)各方面的性能.

**关键词:** 混沌; Pareto 最优; 多目标进化算法

**中图分类号:** TP18 **文献标识码:** A **文章编号:** 0372-2112 (2006) 06-1142-04

## Multi-Objective Chaotic Evolutionary Algorithm

LEI De-ming<sup>1,2</sup>, YAN Xin-ping<sup>1</sup>, WU Zhi-ming<sup>3</sup>

(1. Intelligent Transportation Systems Center, Wuhan University of Technology, Wuhan, Hubei 430063, China;  
2. School of Automation, Wuhan University of Technology, Wuhan, Hubei 430063, China;  
3. Institution of Automation, Shanghai Jiaotong University, Shanghai 200030, China)

**Abstract:** Multi-objective chaotic evolutionary algorithm (MCEA) is designed. In each generation of MCEA, after the population finishes all genetic operations and external archive maintenance is done, chaotic search is performed on the copy of several individuals randomly chosen from the external archive to obtain new non-dominated solutions. MCEAs respectively merging strength Pareto evolutionary algorithm (SPEA), SPEA2 with chaotic search based on Logistic map are applied to some complex multi-objective optimization problems. The computational results demonstrate that the comprehensive performance of multi-objective evolutionary algorithm (MOEA) is improved as a consequence of the inclusion of chaos.

**Key words:** chaos; pareto optimal; multi-objective evolutionary algorithm

### 1 引言

许多实际工程优化问题存在多个彼此冲突的目标函数, 对其中一个目标的优化必须以其他目标为代价, 一个目标的改善使其他目标变劣, 和单目标优化不同, 多目标优化问题的最优解不是唯一的, 而是最优解集合, 是数量众多的 Pareto 最优解的集合. 进化算法具有适合求解多目标优化问题的特点, 首先, 它能同时处理一组解, 每运行算法一次, 就能获得多个有效解, 其次, 进化算法对均衡面 (trade-off surface) 的形状和连续性不敏感, 能很好地逼近非凸或非连续均衡面. 从 Schaffer<sup>[1]</sup> 于 1985 年首次设计向量估计遗传算法 (VEGA) 开始, 多目标进化算法 (MOEA) 经过 1985 ~ 1994 年的停滞期和 1994 年到现在的发展期, MOEA 受到许多研究者的关注.

许多 MOEAs 包括 PAES<sup>[2]</sup>, M-PAES<sup>[3]</sup>, SPEA<sup>[4]</sup>, SPEA2<sup>[5]</sup>, NSGA-2<sup>[6]</sup> 等有一些共同的特征: (1) 构造外部档案保留算法获得的非劣解, 当档案大小超过规定值时, 采取适当措施对档案进行维护; (2) 根据个体间 Pareto 支配关系和密度值, 赋予个体适应度函数值; (3) 很少采用基于适应度值的复制方式如轮盘赌, 而是利用基于局部竞争的复制策略, 其中, 在遗传算法中, 用锦标赛复制, 在进化策略中采用  $\mu_+$  或  $(\mu, \lambda)$  复制.

混沌是非线性系统中的普遍现象, 它对进化算法的应用包括建立基于混沌优化的遗传算子, 利用混沌搜索不断产生更优的最优个体以引导种群进化以及在算法中利用混沌序列替代随机序列. 雷德明<sup>[7]</sup> 利用混沌变量在适应度值最大的几个个体附近搜索新的最优个体, 引导种群进化. Determan 等设计了基于 Logistic 函数的变异算子以增

强算法的 exploitation 能力, Caponetto 等<sup>[9]</sup>研究了在算法中混沌序列完全代替随机数发生器后进化算法性能的改善程度. 李娟<sup>[10]</sup>等利用混沌序列动态地增加种群规模以避免过早收敛. 以上研究主要针对单目标进化算法, 关于混沌搜索对 MOEA 性能的影响, 目前相关研究不多, 本文利用混沌现象的内在特性, 将混沌搜索引入到 MOEA 中, 在每一代, 当种群完成一次进化(复制、交叉和变异)之后, 在外部档案的部分个体附近进行混沌搜索以产生更多非劣解.

## 2 Pareto Optimality

通常, 多目标优化问题数学描述如下.

$$\begin{aligned} \max \quad & f(x) = [f_1(x), f_2(x), \dots, f_m(x)] \\ \text{Subject to} \quad & g_i(x) \leq 0, \quad i = 1, 2, \dots, T, x \in R^b \end{aligned} \quad (1)$$

其中,  $x = (x_1, x_2, \dots, x_n)$ ,  $f_i, 1 \leq i \leq m$  为目标函数,  $g_i$  是约束. 问题具有  $n$  个决策变量,  $m$  个目标函数和  $T$  种约束.

以下是多目标优化中常见的几个基本概念.

(1) Pareto 支配, 解  $x^0$  支配  $x^1 (x^1 > x^0)$  当且仅当

$$\begin{aligned} f_i(x^0) &\leq f_i(x^1), \quad i = 1, 2, \dots, m, \\ f_i(x^0) &> f_i(x^1), \quad \exists i \in \{1, 2, \dots, m\} \end{aligned}$$

(2) Pareto 最优, 如果解  $x^0$  是 Pareto 最优的当且仅当  $\exists x^1: x^1 > x^0$

(3) Pareto-Optimal 集, 所有 Pareto 最优解的集合  $P_s = \{x^0 \mid \exists x^1 > x^0\}$

(4) Pareto-Optimal 前端或均衡面, 所有 Pareto 最优解对应的目标函数值所形成的区域  $P_F$

$$P_F = \left\{ f(x) = (f_1(x), f_2(x), \dots, f_m(x)) \mid x \in P_s \right\}$$

## 3 多目标混沌进化算法 (Multi-Objective Chaotic Evolutionary Algorithm MCEA)

MCEA 与大多数 MOEAs 结构相似, 只是要将混沌搜索插入其中, 其具体步骤如下.

(1) 产生初始种群并将其中的非劣解拷贝到外部档案中

(2) 从种群和外部档案中选择个体进入下一代

(3) 执行交叉、变异操作

(4) 计算种群中各个体对应的目标函数向量并维护档案

(5) 从外部档案中随机选择几个个体, 分别以这些个体的拷贝作为初始解, 进行混沌搜索并根据混沌搜索产生的非劣解进行档案维护

(6) 如果满足终止条件, 停止搜索; 否则, 转到(2).

下面介绍混沌搜索的具体过程.

混沌优化<sup>[11,12]</sup>利用混沌变量的随机性、遍历性等内在特性, 它的基本思想是把混沌变量线性映射到优化变量的取值区间, 然后利用混沌变量进行搜索. 混沌优化本质上还是一种局部搜索方法, 它必须在搜索过程中不断缩小优

化变量的搜索区间, 不然算法很难获得最优解. 本文利用最常见的混沌映射-Logistic 映射

$$u_{k+1} = u_k(1 - u_k) \quad (2)$$

$u_0 < 1$ , 当  $\mu = 4$ , Logistic 系统处于完全混沌状态.

外部档案已成为许多 MOEA 不可分割的部分, 它主要用来保留算法在搜索过程中获得的非劣解, 档案被规定了最大规模, 当档案内解的个数达到规定值时, 要对档案进行维护, 以决定新解能否加到档案中, 以及如果新解加到档案后, 从其中移出哪些解以确保档案大小始终不超过规定值. 当混沌优化与 MOEA 结合时, 仅从外部档案中随机选择几个解, 以这些解的拷贝作为混沌搜索的初始解, 在它们所在的附近区域内搜索新的非劣解. 由于只是档案个体的拷贝参与混沌搜索, 没有档案个体从档案中移出. 混沌搜索产生的非劣解进入外部档案, 并根据 MOEA 的相应方法对档案进行维护.

混沌搜索基本步骤如下

(1) 设定  $u_{0j}, j = 1, 2, \dots, n, k = 0, p_{cs}, n_{cs}, cs_{max}, h = 1$

(2) 对每个决策变量确定  $[j_{min}, j_{max}]$ , 其中  $j_{min} = \min_{x \in P} \{x_j\}$ ,  $j_{max} = \max_{x \in P} \{x_j\}$ , 如果  $j_{min} - 0.1 \leq j_{max} - j_{min} \leq j_{max}$ , 则  $j_{min} = j_{max}$ ; 否则  $j_{min} = j_{min} - 0.1$ , 如果  $j_{max} + 0.1 \leq j_{max} - j_{min} \leq j_{max}$ , 则  $j_{max} = j_{min}$ ; 否则  $j_{min} = j_{min} + 0.1$

(3) 随机地从档案中选择一个解, 对其拷贝  $x^h = (x_{h1}, x_{h2}, \dots, x_{hn}), v = 1$

(a) 对每个  $x_{hj}, j = 1, 2, \dots, n$ , 产生一个满足均匀分布  $U(0, 1)$  的随机数  $s$ , 若  $s < p_{cs}$ , 则  $y_{hj} = x_{hj} + c_j u_{kj} - d_j$ ; 否则  $y_{hj} = x_{hj}$

(b) 计算新解  $y_v = (y_{v1}, y_{v2}, \dots, y_{vn})$  对应的目标函数向量, 如果它不被种群和档案中的任何解支配, 将其保留在集合中,  $k = k + 1, v = v + 1$ , 若  $v < cs_{max}$ , 转到 (a); 否则转到 (4)

(4) 如果解集  $\Omega$  中所有非劣解不能支配档案中的任何个体, 将  $\Omega$  中一个非劣解加入到档案中并进行档案维护; 否则, 对于  $\Omega$  中的每个非劣解, 若它支配了档案中的部分解, 让它替代档案中的那些受其支配的个体.

(5)  $h = h + 1$ , 若  $h < n_{cs}$ , 则清空集合  $\Omega$  并转到 (3); 否则停止搜索

其中,  $u_{kj}, j = 1, 2, \dots, n$  是对应于第  $j$  个决策变量的 Logistic 混沌变量,  $u_{0j}$  为其对应的初值.  $p_{cs}$  是每个决策变量的混沌搜索概率而  $cs_{max}$  则是每个被选中的档案个体的拷贝进行混沌搜索的最大次数.  $n_{cs}$  为档案中被选中的解的个数.  $[j_{min}, j_{max}]$  为第  $j$  个决策变量的定义域.  $c_j, d_j$  按如下原则确定以确保每个决策变量在区间  $[j_{min}, j_{max}]$  内变化:

对于  $x_{hj} \in [j_{min}, j_{max}]$ , 若  $j_{min} - x_{hj} > j_{max} - j_{min}$ , 则  $t_1 = j_{min} - x_{hj}$ . 若  $x_{hj} - j_{min} > j_{max} - j_{min}$ , 则  $t_2 = j_{max} - x_{hj}$ . 最终  $c_j = t_1 + t_2, d_j = t_2 \cdot j_{min}, j = 1, 2, \dots, n$  为小正数.

混沌搜索产生的大多数解难以成为外部档案的成员除非它们支配档案内的个体, 这样可以避免档案中的解过

于相似,因为混沌搜索仅改变档案个体的拷贝的部分基因位,并且变化幅度比较小.

4 计算结果

多目标优化的目的有两个:(1)收敛到 Pareto-Optimal 集;(2)算法获得的非劣解均匀地分布在整个均衡面上. Zitzler, et al 等<sup>[13]</sup>给出了 3 个指标分别描述算法是否实现了上述两目的.

$$M_1^*(Y) = \frac{1}{|Y|} \min \left\{ p - \bar{p}; \bar{p} - \bar{Y} \right\} \quad (3)$$

$$M_2^*(Y) = \frac{1}{|Y|} \left\{ q - Y, q - p^* > \right\} \quad (4)$$

$$M_3^*(Y) = \sqrt{\max_{i=1}^m \left\{ p_i - q_i^*, p, q \right\}} \quad (5)$$

其中  $Y$  为算法获得地非劣解集而  $\bar{Y}$  是问题的 Pareto-Optimal 前端.  $M_1^*(Y)$  确定  $Y$  与  $\bar{Y}$  之间的平均距离.  $M_3^*(Y)$  测量  $Y$  中的非劣解在均衡面上的分布范围,而分布指标  $M_2^*$  则反映了  $n$ -niches 的数量,其值取在区间  $[0, |Y|]$  内.

表 1 MCEA 参数

ZDT4	$i_1=0.2, j=0.6, j=2, \dots, n,$ $p_{cs}=0.2$ MCEA2 $i_1=0.15, j=0.6, j=2, \dots, n, p_{cs}=0.2$ MCEA1	$n_{cs}=1$ cs. max=5	26500*
ZDT6	$i_1=0.1, j=1, 2, \dots, n, p_{cs}=0.2$	$n_{cs}=1$ cs. max=5	26500
GWK	$i_1=0.2, j=1, j=2, \dots, n,$ $p_{cs}=1$ MCEA2 $i_1=0.2, j=0.8, j=2, \dots, n,$ $p_{cs}=0.2$ MCEA1	$n_{cs}=1$ cs. max=5	26500
DTLZ3	$j=0.2, j=1, 2, \dots, n, p_{cs}=0.2$	$n_{cs}=2$ cs. max=5	105000
DTLZ6	$j=0.2, j=1, 2, \dots, n, p_{cs}=0.2$	$n_{cs}=2$ cs. max=5	105000

\*第 4 列中的数据为各算法求解相应问题时,目标函数向量的计算次数.

表 2 所选指标的均值(第一行)和方差(第二行)

指标	算法	ZDT4	ZDT6	GWK	DTLZ3	DTLZ6
$M_1^*$	SPEA	2.3645	0.054387	0.087303	1.36883	0.10966
		7.1142	0.003561	0.017459	2.9782	0.00621
	MCEA1	0.168706	0.04608	0.05409	0.35790	0.08886
		0.04932	0.0015	0.00223	0.261202	0.000696
	SPEA2	0.11574	0.011537	0.057028	0.34374	0.066425
		0.006889	0.000004	0.00457	0.17462	0.000147
MCEA2	0.04255	0.010594	0.03001	0.04841	0.06540	
	0.002219	0.000006	0.00233	0.000226	0.00001	
$M_2^*$	SPEA	13.899	20	18.36732	49.4286	49.298
		8.0276	0	2.2051	0.3007	0.03281
	MCEA1	20	20	19.7667	49.289	50
		0	0	0.24266	0.16876	0
	SPEA2	19.7474	20	19.5	50	50
		0.18208	0	0.94444	0	0
MCEA2	20	20	19.5	50	50	
	0	0	0.5	0	0	

(续表 2)

指标	算法	ZDT4	ZDT6	GWK	DTLZ3	DTLZ6
$M_3^*$	SPEA	0.6971	1.4366	0.732316	5.452612	3.3657
		0.07675	0.94199	0.051265	2.2276	0.33613
	MCEA1	1.03343	1.35261	1.05631	3.29707	3.7586
		0.04347	0.54961	0.03296	2.67573	0.28024
	SPEA2	0.985108	1.1665	1.06665	4.8517	3.77489
		0.086052	0.00003	0.04689	7.8362	0.09649
	MCEA2	1.06499	1.16398	1.28952	1.79476	3.95964
		0.03193	0.000006	0.022185	0.001078	0.07616

5 个实例用于测试混沌搜索对 MOEA 性能的影响. 其中 ZDT4<sup>[13]</sup>具有 21<sup>9</sup> 局部 Pareto-Optimal 前端而其中只有一个最优. GWK<sup>[14]</sup>是 ZDT4 的变形,将 ZDT4 中的函数改成 Greigwank 函数即可. GWK 具有 163<sup>9</sup> 个局部 Pareto-Optimal 前端. ZDT6<sup>[13]</sup>也是一个常见的测试实例. DTLZ3<sup>[15]</sup>具有 3 个目标函数、12 个决策变量和 3<sup>9</sup> 个局部 Pareto-Optimal 前端. DTLZ6<sup>[15]</sup>是个 22 维的 3 目标优化问题,其 Pareto-Optimal 前端在目标空间上不连续,一共有 4 块 Pareto-Optimal 曲面.

混沌搜索分别与 SPEA 和 SPEA2 混合,形成 MCEA1 和 MCEA2,这两种 MOEAs 采用实数编码策略,二元锦标赛选择、模拟二进制交叉(SBX)  $p_c=20$  和多项式变异(Polynomial Mutation)  $p_m=20$ . 对于 2 目标问题,种群规模  $N=80$ ,外部档案最大规模  $N=20$ ,对于 3 目标问题,  $N=150, N=50$ . 对于所有问题,交叉概率  $p_c=0.9$ ,变异概率  $p_m=0.1$ ,MCEA 其它参数见表 1,表中参数若没特殊说明,同时适用于 MCEA1 和 MCEA2. 每个算法对每个测试问题运行 20 次. 计算结果如表 2 所示,图 1~2 描述了 3 个测试实例的对比结果,图中,实线表示问题的真实最优前端, MCEA1 对应的 MOEA 是 SPEA, MCEA2 对应的 MOEA 为 SPEA2.

从表 2 可以看出,混沌搜索加入 MOEA 后,算法的收敛性能得到改善,对于 3 个具有很多局部 Pareto-Optimal 前端的问题,混沌加入前后,  $M_1^*$  值变化明显,如图 1 所示, SPEA 很难逼近 ZDT4 的全局 Pareto-Optimal 前端,引入混沌后, MCEA1 搜索到的非劣解距全局 Pareto-Optimal 前端的距离大大减少了,而 MCEA2 也比 SPEA2 更好地逼近这些问题的均衡面. 对于 ZDT6 和 DTLZ6,由于问题相对比较容易求解,混沌加入后,  $M_1^*$  值变化幅度不大. 出现上述结果的主要原因在于混沌搜索的加入明显降低了 MCEA 陷入局部最优解的可能性,避免了过早收敛.

由于混沌搜索不断产生新的非劣解并加入到外部档案中,这样提高了档案内解的多样性,同时也使遗传搜索在更广的区域内进行,使 MCEA 产生的非劣解更加均匀地分布在更大范围的均衡面上,如表 2 所示,对于所有问题,混沌加入后, MCEA 都获得了比对应的 MOEA 更好的  $M_3^*$ ,例如, MCEA1 关于 ZDT4 获得的  $M_3^*$  值优于 SPEA 的对应结果. 关于 DTLZ3,尽管 SPEA 和 SPEA2 对应的

$M_3^*$  都比 MCEA1 和 MCEA2 大,但后者对应的结果更有效,因为它们更接近最优结果;随着 MCEA 搜索到的解的分布范围变优,解的分布均匀性并没有降低,对于有些问题,均匀性也得到了改善。

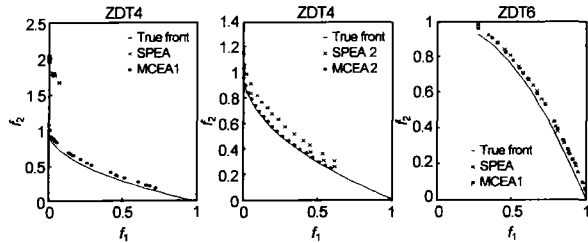


图 1 MCEA 与对应的 MOEA 关于 ZDT4 和 ZDT6 的对比结果

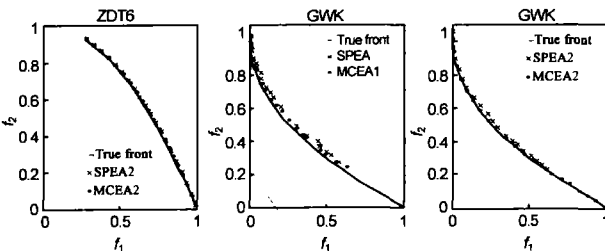


图 2 MCEA 与对应的 MOEA 关于 ZDT6 和 GWK 的对比结果

可以发现,混沌对算法性能的影响与算法本身的性能有关,如果算法本身对某些问题取得了很好的结果,混沌的作用就很小,例如,SPEA2 关于 ZDT6 和 DTLZ6 的计算结果和 MCEA2 的很接近。而对于 ZDT4 等问题,MOEA 的计算结果比较差,混沌的引入,能明显地改善 MOEA 各方面地性能,使 3 个性能指标同步变好。

## 5 结论

本文研究了混沌和 MOEA 的结合问题,将混沌引入到 MOEA 中,在算法每一代,从外部档案中选择部分个体并在这些个体的拷贝所在的附近区域进行混沌搜索以产生更多非劣解。将混沌与 SPEA、SPEA2 结合的混合算法应用于多个高维且具有很多局部 Pareto-Optimal 前端的测试实例,计算结果表明,混沌的加入,明显改善了 MOEA 各方面的性能。

## 参考文献:

- [1] Schaffer J D. Multiple objective optimization with vector evaluated genetic algorithm [A]. Proc 1st International Conference on Genetic Algorithm [C]. Mahwah, NJ, USA:Lawrence Erlbaum Associates, 1985. 93 - 100.
- [2] Knowles J D, Corne D W. Approximating the non-dominated front using the pareto archive evolutionary strategy [J]. Evolutionary Computation, 2000, 8(2): 149 - 172.
- [3] Knowles J D, Corne D W. M-PAES. A multi-objective memetic algorithm [A]. Proc 2000 Congress on Evolutionary Computation [C]. Piscataway, NJ: IEEE Press, 2000. 325 - 332.
- [4] Zitzler E, Thiele L. Multi-objective evolutionary algo-

rithms: a comparative case study and the strength Pareto approach [J]. IEEE Transaction on Evolutionary Computation, 1999, 3(4): 257 - 271.

- [5] Zitzler E, Laumanns M, Thiele L. SPEA2: Improving the strength Pareto evolutionary algorithm [R]. (TIK-Rep 103) Lausanne, Switzerland: Swiss Federal Institute of Technology, 2001. 1 - 21.
- [6] Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multi-objective genetic algorithms: NSGA-II [J]. IEEE Transactions on Evolutionary Computation, 2002, 6(2): 182 - 197.
- [7] 雷德明. 利用混沌搜索全局最优解的一种混合遗传算法 [J]. 系统工程与电子技术, 1999, 21(12): 81 - 82.
- [8] Determan J, Foster J A. Using chaos in genetic algorithm [A]. Proceedings of the 1999 congress on Evolutionary Computation [C]. Piscataway, NJ: IEEE Press, 1999. 2094 - 2101.
- [9] Caponetto R, Fortuna L, Fazzino S, et al. Chaotic sequence to improve the performance of evolutionary algorithms [J]. IEEE Transactions on Evolutionary Computation, 2003, 7(3): 289 - 304.
- [10] Juan L, Zixing C, Jianqin L. Premature convergence in genetic algorithm: analysis and prevention based on chaos operator [A]. Proc 3rd World Congress Intelligent Control Automation [C]. USA: IEEE Press, 2000. 495 - 499.
- [11] 张彤, 王宏伟, 王子才. 变尺度混沌优化方法及其应用 [J]. 控制与决策, 1999, 14(3): 285 - 287.
- [12] 李兵, 蒋慰孙. 混沌优化方法及其应用 [J]. 控制理论与应用, 1997, 14(4): 613 - 615.
- [13] Zitzler E, Deb K, Thiele L. Comparison of multi-objective evolutionary algorithm: empirical study [J]. Evolutionary Computation, 2000, 8(2): 173 - 195.
- [14] Deb K. Multi-objective genetic algorithm: problem difficulties and construction of test function [J]. Evolutionary Computation, 1999, 7(3): 205 - 230.
- [15] Deb K, Thiele L, Zitzler E. Scalable multi-objective optimization test problems [A]. Proc 2002 Congress on Evolutionary Computation [C]. Piscataway, NJ: IEEE Press, 2002. 825 - 830.

## 作者简介:



雷德明 男, 1968 年 5 月出生于湖北石首, 主要研究方向: 多目标进化算法及其应用, 多目标微粒群算法等。E-mail: deminglei11@163.com