

基于 ϵ 占优的正交多目标差分演化算法研究

龚文引, 蔡之华

(中国地质大学计算机学院 武汉 430074)

(cug11100304@163.com)

Research on ϵ -domination based Orthogonal Differential Evolution Algorithm for Multi-objective Optimization

Gong Wenyin, and Cai Zhihua

(School of Computer Science, China University of Geosciences, Wuhan, 430074, China)

Abstract: Evolutionary multi-objective optimization (EMO) has become a very popular topic in the last few years. However, to design an efficient and effective EMO algorithm to find the *near-optimal* and *near-complete* Pareto front is a challenging task. In this paper, we propose a novel differential evolution algorithm to solve multi-objective optimization problems (MOPs) efficiently. The proposed approach uses an Archive population to retain the obtained non-dominated solutions; also it adopts the orthogonal design method with quantization technique to generate an initial population of points that are scattered uniformly over the feasible solution space, so that the algorithm can evenly scan the feasible solution space once to locate good points for further exploration in subsequent iterations. Moreover, it is based on the ϵ -dominance concept to obtain a good distribution of Pareto-optimal solutions and gets them in a small computational time. To make the algorithm converge faster, the new approach employs a hybrid selection mechanism in which a random selection and an elitist selection are interleaved. Experiments on eight benchmark problems of diverse complexities show that our approach is able to obtain a good distribution in all cases. Compared with several other state-of-the-art evolutionary algorithms, it achieves not only comparable results in terms of convergence and diversity metrics, but also a considerable reduction of the computational effort. Furthermore, we discuss the influence of different CR value and the parameter value of hybrid selection mechanism to the performance of the algorithm experimentally.

Key words: Multi-objective optimization; differential evolution algorithm; orthogonal design method; ϵ -domination; hybrid selection

摘要: 演化多目标优化是目前演化计算中热门研究方向之一。但是,要设计一种高效、鲁棒的演化多目标优化算法使其找到接近最优和完整的非劣解集是一项很困难的任务。为了能有效求解多目标优化问题,提出了一种新的多目标差分演化算法。新算法具有如下特征:1)利用正交实验设计和连续空间量化的方法产生初始群体,使得初始群体中的个体可以均匀分布于搜索空间,并且可以使好的个体在演化过程中得到利用。2)采用 Archive 群体保存非劣解,并利用 ϵ 占优方法更新 Archive 群体,从而可以使算法较快获得分布很好的 Pareto 解集。3)为了加快算法收敛,提出一种基于随机选择和精英选择的混合选择机制。通过 8 个标准测试函数对新算法进行测试,并与其他一些多目标演化算法进行比较,其结果表明新算法可以有效逼近真实 Pareto 前沿且分布均匀,并且在收敛性和多样性的求解精度和稳定性上与其他算法相当。同时,通过实验的方法验证了新算法改进之处的有效性,并进一步讨论了差分演化算法中 CR 取值和混合选择机制中选择控制参数 λ 取值对算法性能的影响。

关键词: 多目标优化; 差分演化算法; 正交实验设计; ϵ 占优; 混合选择

中图法分类号: TP391.9

演化算法是一种基于群体的启发式搜索算法,它利用群体中的个体在解空间中进行搜索,具有自适应、自学习、自组织和隐并行性等特点。目前,利用演化算法求解多目标优化问题(Multi-objective Optimization Problems, MOPs)已成为演化计算一个热门研究方向^[1]。与传统求解 MOPs 的数值算法不同,多目标演化算法(Multi-objective Evolutionary Algorithms, MOEAs)可以在每次运行中得到一组非劣解^[2],从而有利于决策者根据求解问题的需要从中选择最合适的一个解作为问题的最终解。为了使决策者便于选择合适解,要求算法所得到的非劣解集满足以下两个特点: i)所得非劣解集尽可能接近真正 Pareto 前沿; ii)所得非劣解集要尽可能均匀分布。这两个特点构成了 MOEAs 的两个目标^[3],很多

算法的评价标准也是根据这两个目标设计的。

在过去的几十年里,一些学者提出了多种基于演化算法的多目标优化算法,如 NSGA-II^[2]、SPEA2^[4]、PAES^[5]、GDE3^[6]、MOEO^[27]等。在这些算法中,以 Deb 等提出的 NSGA-II 和 Zitzler 等提出的 SPEA2 最为出名,许多新提出的演化多目标算法都与这两个算法的实验结果进行比较来验证新算法的有效性。

近年来,国内一些学者在这方面进行了大量的研究。曾三友等^[7]把正交实验设计的方法应用到多目标问题求解中,利用正交数组产生小生境群体,并利用统计优化方法繁殖后代,从而加快收敛速度。石川等^[8]提出一种基于占优树的多目标进化算法,算法利用占优树进行适应值分配,减少了个体间的比较。张利彪等^[9]提出了基于极大极小距离密度

收稿日期:

修回日期:

基金项目:“十一五”民用航天项目(C5220061318);湖北省人文基地项目(2004B0011);湖北省自然科学基金(2003ABA043)。

的多目标差分进化算法,采用极大极小距离密度来保持非劣解集.崔逊学等^[10]提出基于偏好的多目标调和遗传算法,利用偏好信息对个体进行选择.

尽管目前已经有了很多演化多目标算法,但是,要设计高效、鲁棒的演化多目标算法使其找到接近最优和完整的非劣解集是仍一项很困难的任务.为了能有效求解多目标优化问题及增强多目标差分进化算法的鲁棒性,笔者对差分进化算法进行改进,提出了一种基于 ϵ 占优的正交多目标差分进化算法,称之为 ϵ -ODEMO.新算法的主要改进之处是:1)采用正交实验设计和连续空间量化的方法来产生初始 Archive 群体和演化群体,使群体中的个体均匀分布在整个搜索空间中,并且一旦产生好的个体,可以在随后的演化中得到利用,增强了算法的利用能力;2)采用 Archive 群体保存得到的非劣解集并用 ϵ 占优方法对其进行更新,使算法能较快得到分布很好的非劣解集,并确保精英解不会在演化中丢失;3)为了加快算法收敛速度和利用 Archive 群体中的个体指导演化,采用一种基于随机选择和精英选择的混合选择机制.最后,通过实验表明 ϵ -ODEMO可以有效逼近真实 Pareto 前沿且分布均匀,在求解精度和稳定性上与所比较算法相当;并通过实验的方法验证了算法改进之处的有效性.

1 相关研究

1.1 多目标差分进化算法

差分进化算法^[11]是一种简单、有效、快速的演化算法,已在优化领域得到了广泛的应用^[12].目前,一些学者把 DE 算法应用到多目标优化问题的求解中,提出了一些多目标差分进化算法.如 Robi \acute{c} 等^[3]等提出的 DEMO,该算法采用一个子群体来保存当子个体和父个体非劣时的子个体,并且子群体非空时把子群体与演化群体合并,利用非劣解排序和拥挤距离对合并的群体进行裁剪,把裁剪后的群体作为下一代的演化群体. DEMO 结合 DE 算法的优点,把 NSGA-II 中的非劣解排序和拥挤距离引入到算法中,使算法可以较快收敛于真正 Pareto 前沿. Kukkonen 等^[6]提出的 GDE3,该算法对 GDE^[15]算法的改进,增强了算法对约束函数处理的能力. Iorio 等^[13]提出了 NSDE 算法,并把它应用的旋转 MOPs 的求解中,该算法与 NSGA-II 的唯一区别在于采用 DE 算子代替 NSGA-II 里面的交叉和变异算子. Abbass 等^[14]提出了 PDE 算法,采用高斯随机数产生初始群体,并设计了自适应交叉变异算子. 张利彪等^[9]提出了基于极大极小距离密度的多目标差分进化算法,采用极大极小距离密度来保持非劣解集.但是,这些算法都采用随机初始群体,降低了初始群体的利用能力,而且只采用一个演化群体,使算法的选择压力过大,不能有效利用非劣解指导演化;此外,目前把 DE 算法应用到 MOPs 中一般都只采用 Pareto 占优方法比较解的优劣,没有采用文献^[20]中提出的 ϵ 占优方法保存非劣解.这些都是本文研究的动机.

1.2 正交设计在 EA 中的应用

在工农业和科学研究中,经常要考虑多个因素对产品的影响.如果对每个因素不同水平的相互搭配安排全面实验的话,常常是很困难甚至不可能实现的.正交设计法^[16]是一种通过较少数次试验,就能找到最好或较好的试验条件.因此,它被广泛地用于寻优.在寻优时,先确定影响指标的因素和水平,再选择适当的正交表,即可按正交表安排试验,最后,分析试验的结果和发现较好的水平.

Leung 等^[17]把正交实验设计方法应用到遗传算法中,并给出了产生正交矩阵的算法,通过正交设计产生初始群体,并设计了正交杂交算子,通过测试验证了算法的有效性. Gong 等^[18]把二水平正交杂交算子应用到差分进化算法中,并结合统计优生法从杂交子群体中选择最优个体,增加了算法的鲁棒性.曾三友等^[7]把正交实验设计的方法应用到多目标问题求解中,利用正交数组产生小生境群体,并利用统计优化方法繁殖后代,从而加快收敛速度.但是,该算法的缺点是演化每一代中通过正交设计产生小生境,增加了算法的复杂度.因此,本文对正交设计在 MOPs 中的应用作了进一步研究.

1.3 基于 ϵ 占优方法的 MOEAs

为了设计一种能快速得到均匀分布的非劣解集的多目标演化优化, Laumanns 等^[19]采用一种宽松的 Pareto 比较方法(ϵ 占优)来比较个体的优劣,并利用第二个群体 Archive 来保存非劣解,通过把目标空间划分成很多网格,且规定每个格子只能保存一个非劣解,保证算法可以在较短的时间内得到分布较好的非劣解集.此方法提出后,许多学者把此方法应用到 MOEAs 中^[20-22],得到了较好的结果.采用 ϵ 占优方法优点是算法可以在较短时间得到均匀分布的非劣解集,且工程人员可通过控制 ϵ 值的大小来获得指定数目的非劣解而无须预先设置 Archive 群体的大小.此外,文献^[19]指出在 ϵ 占优方法中如何有效利用 Archive 中的个体指导演化值得进一步研究.文献^[20,21]在每一代演化中采用随机方法从 Archive 中选择一个个体参与重组.在文献^[22]中所有的子个体均利用 Archive 中个体产生.由于这两种方法过多利用 Archive 中的个体信息,因此,当 Archive 中包含很多无效个体时有可能导致演化,使算法陷入局部 Pareto 前沿.

2 改进差分进化算法

2.1 正交初始群体

传统的演化算法一般都采用随机方法来产生初始群体,这样产生的群体分布不均匀,很难产生好的个体,降低了算法对初始群体的利用能力.采用正交实验设计方法产生初始群体,可以使群体中的个体均匀地分布于整个搜索空间,而且可以产生好的个体,这些个体可以增强算法在随后演化过程中的利用能力,加速算法的收敛^[17].为了增强算法的有效性和加快算法收敛,本研究中采用正交实验设计的方法产生初始 Archive 群体和演化群体.

2.1.1 产生正交数组

在产生正交初始群体之前,算法先要根据不同的问题

产生不同的正交矩阵(Orthogonal Array, OA, 产生 OA 的详细算法请参考文献[17]), 在本文的研究中用 $L_R(Q^C)$ 表示具有 Q 个不同水平的 OA, 这里 Q 为奇数, $R=Q^J$ 表示 OA 的行数, 正交指数 J 是一个正整数, 满足以下条件:

$$C = (Q^J - 1)/(Q - 1) \quad (1)$$

C 表示 OA 的列数, 为了使所产生的 OA 能满足正交初始化群体的需要, C 必须大于或等于优化问题自变量的个数. 因此, 要设计一个合适的 OA 则满足(一般, 为了减少初始群体的评价次数, 取指数 $J=2, Q$ 取满足下面条件的最小值即可):

$$R = Q^J \geq N \quad \text{s.t.: } C \geq n \quad (2)$$

其中, n 表示所要优化问题的自变量个数, N 表示演化群体的大小.

2.1.2 搜索空间量化

为了能将正交实验设计的方法应用到连续空间中, 需要对连续空间进行量化. 假设优化问题某一自变量的范围是 $[l, u]$, 将其量化为 Q 个水平 $\alpha_1, \alpha_2, \dots, \alpha_Q$, 其中 Q 为奇数, α_i 由如下公式计算:

$$\alpha_i = l + (i-1)((u-l)/(Q-1)), \quad i = 1, \dots, Q \quad (3)$$

这样 $[l, u]$ 连续空间就被量化成 $Q-1$ 个相等的部分.

2.1.3 产生初始 Archive 群体及初始演化群体

根据所优化的问题按公式(2)产生了合适的 OA 和对搜索空间量化之后, 则可根据 OA 产生正交群体 OP(具体方法请参考文献[17]), 由公式(2)可知, 此时 OP 的个体数目大于演化群体的种群数 N , 为此, 在本文研究中, 先通过 Pareto 占优的方法从 OP 中选择非劣个体, 形成初始 Archive 群体, 设此时 Archive 群体大小为 ar_size . 产生完初始 Archive 群体后, 再产生初始演化群体. 如果 $ar_size \geq N$, 则从 Archive 群体中随机选择 N 个个体作为初始演化群体; 否则, 除把 Archive 群体中的所有个体加入到初始演化群体外, 剩下 $N-ar_size$ 个个体从正交群体中随机选取. 与文献[17]比较主要区别是: 1)将正交群体初始化应用于 EMO 中, 根据 Pareto 占优从 OP 中产生初始 Archive 群体; 2)由初始 Archive 群体和 OP 共同产生演化初始群体. 值得指出的是对于指定 Q 和 J , 算法每次产生的 OA 是固定的, 因此可以采用离线方式产生 OA, 避免了文献[7]采用正交设计产生小生境而增加算法的复杂度.

2.2 混合选择机制

在传统 DE 算法中, 采用如下差分方程对个体进行重组

$$X' = X_{r1} + F \times (X_{r2} - X_{r3}) \quad (4)$$

其中, X_{r1} 称为基点(Base point), $r1 \neq r2 \neq r3$ 是从演化群体中随机选择的三个父个体. 为了加快算法收敛速度和利用 Archive 群体中的个体指导演化, 在 ϵ -ODEMO 中采用基于随机选择和精英选择的混合选择机制.

$$\text{selection} = \begin{cases} \text{random} & \text{eval} < (\lambda \times \text{Max_eval}) \\ \text{elitist} & \text{otherwise} \end{cases} \quad (5)$$

其中, $eval$ 是当前适应值评价次数, Max_eval 是最大评价次数, $\lambda \in [0.1, 1.0]$ 为选择机制控制参数. 在随机选择中, 三个父个体从演化群体中随机选取, 而在精英选择中, 基点 X_{r1} 从 Archive 群体中随机选取, 其他两个个体则从演化群体中随机选取. 这是因为在演化初期, Archive 群体中的个体大部分不是最终的非劣解, 对演化的指导作用小; 随着演化的进行, Archive 群体中的好个体数目增加, 利用其中的个体作为基点, 可以加快算法收敛到全局最优解的速度. 与文献[22]中方法不同之处是本算法中精英选择只把 Archive 中个体作为基点; 而[22]中所有父个体均从 Archive 中选取, 当 Archive 中包含很多无效个体时有可能不会引导演化; 且由于所有父个体均从 Archive 中选取使演化群体失去意义.

2.3 非劣解存储

为了能够在较短的时间内获得分布均匀的非劣解集, 本文采用文献[20]中提出的 ϵ 占优方法更新 Archive 群体, 存储非劣解. 每个解在 Archive 中分配一个识别数组 $B=(B_1, B_2, \dots, B_{nobj})$, 其中 $nobj$ 为目标函数个数:

$$B_j(f) = \lfloor (f_j - f_j^{\min}) / \epsilon_j \rfloor \quad \text{for minimizing } f_j \quad (6)$$

其中, f_j^{\min} 是第 j 个目标函数的可能最小值; ϵ_j 是第 j 个目标函数容差. 利用识别数组 B 把整个目标空间划分成许多网格, 每个解根据相应的识别数组判断优劣, 按 ϵ 占优方法存储在网格中, 并规定每个网格只能存放一个解. Archive 更新过程算法 1 所示.

2.4 新算法流程

ϵ -ODEMO 算法的流程大致与文献[3]的 DEMO 算法流程相似, 其区别主要是 ϵ -ODEMO 采用正交实验方法产生初始群体, 利用 Archive 群体保存非劣解并通过 ϵ 占优的方法更新 Archive 群体; 当子群体非空时, 把子群体与演化群体合并, 对合并群体的裁剪只使用非劣解排序的方法, 而用随机选取的方法代替拥挤距离选取. 这是因为, ϵ -ODEMO 算法所得到的非劣解集通过 Archive 群体保存, 而 Archive 群体的多样性通过 ϵ 占优方法保持, 因此, 演化群体的多样性就可以不考虑. 此外, ϵ -ODEMO 算法采用了混合选择机制, 利用 Archive 群体中个体的信息指导演化, 加快了算法的收敛. 其算法流程如算法 2 所示. 其中 $eval$ 表示函数评价次数, 随目标函数被评价而增加; Max_eval 是最大函数评价次数, 当 $eval \geq Max_eval$ 时循环终止; $P(j)[i]$ 表示第 j 代群体 $P(j)$ 中的第 i 个个体; 限于篇幅 DE/rand/1/bin 策略本文不再详述, 感兴趣读者请参考文献[11], 并可从[12]中获得 DE 算法的源程序.

Algorithm 1: Updating of Archive with ϵ -dominance

```
If the offspring  $C$   $\epsilon$ -dominates some solutions in Archive then
    Delete all solutions  $\epsilon$ -dominated by  $C$  in Archive
    Insert  $C$  into Archive
Else if  $C$  is  $\epsilon$ -dominated by any solution in Archive then
    Reject  $C$ 
Else //  $C$  is  $\epsilon$ -nondominated with respect to Archive solutions
    If  $C$  shares the same grid with an Archive solution  $D$  then
        If  $C$  usual-dominates  $D$  or  $C$  is closer to the grid than  $D$  then
            Delete  $D$  from Archive and accept  $C$ 
        Else
            Reject  $C$ 
        End if
    Else //  $C$  does not share the same grid with any Archive solution
        Insert  $C$  into the Archive
    End if
End if
```

Algorithm 2: Procedure of the proposed ϵ -ODEMO

```
Generate a proper OA and generate the orthogonal population  $OP$ 
Create the initial Archive  $E(0)$  with  $\epsilon$ -nondominated solutions from  $OP$ 
Create the initial orthogonal evolutionary population  $P(0)$  from  $E(0)$  and  $OP$ 
while  $eval < Max\_eval$  do
    for  $i=1$  to  $N$ 
        if  $eval < \lambda \times Max\_eval$  then
            Random selection
            Produce the new solution with DE/rand/1/bin strategy
        else
            Elitist selection
            Produce the new solution with DE/rand/1/bin strategy
        end if
        if the new solution dominates the target solution  $P(j)[i]$  then
            Replace the target solution with the new solution
        else if the new solution is non-dominated by the target solution  $P(j)[i]$  then
            Add the new solution in the child population  $CP$ 
             $child\_size++$ 
        else
            Discard the new solution
        end if
        Update the Archive  $E(j)$  with  $\epsilon$ -dominance concept
    end for
    if the child population is not empty then
        Combine  $CP$  and  $P(j)$ 
        Prune the mixed population using non-dominated ranking method only
        Get the next evolutionary population  $P(j+1)$ 
    end if
end while
```

Fig. 1 Procedure of the proposed ϵ -ODEMO

图 1. ϵ -ODEMO 算法流程图

3 算法测试及结果分析

3.1 测试函数

为了验证改进算法的性能,选择 8 个标准测试函数(见表 1)作为测试用例,其中 6 个 ZDT 函数只有 2 个目标函数;前 5 个函数(ZDT1-4, ZDT6)来自文献[23],这 5 个函数是变量无关的函数,第 6 个函数 ZDT1.1^[24]是 ZDT1 函数的变形,它是一个变量相关的函数,只有当所有变量与第一个变量的值相等时,才能收敛到真正 Pareto 前沿.DTLZ1 和 DTLZ7 来自文献

[26]具有 3 个目标函数,在本文的研究中 DTLZ1 的自变量个数为 12 个,这增加了此函数的求解难度.

3.2 评价标准

为了说明算法的有效性并与其他算法进行比较,本文采用文献[2]提出的两个评价标准.假设 Q 为算法所得到的非劣解集, P^* 为问题的真正 Pareto 前沿近似集.

1) 收敛性 γ . 收敛性用来评价 Q 与 P^* 之间的逼近程度:

$$\gamma = \left(\sum_{i=1}^{|Q|} d_i \right) / (|Q|) \quad (6)$$

Table 1 Benchmark problems used in this research

表 1 实验测试函数

Problem	n	Domain	Objective functions	Comments
ZDT1	30	$[0, 1]^n$	$f_1(x) = x_1, f_2(x) = g(x)[1 - (x_1/g(x))^{0.5}]$ $g(x) = 1 + 9(\sum_{i=2}^n x_i)/(n-1)$	high dimensionality, convex
ZDT2	30	$[0, 1]^n$	$f_1(x) = x_1, f_2(x) = g(x)[1 - (x_1/g(x))^2]$ $g(x) = 1 + 9(\sum_{i=2}^n x_i)/(n-1)$	high dimensionality, non-convex
ZDT3	30	$[0, 1]^n$	$f_1(x) = x_1, f_2(x) = g(x)[1 - \sqrt{x_1/g(x)} - x_1 \sin(10\pi x_1)/g(x)]$ $g(x) = 1 + 9(\sum_{i=2}^n x_i)/(n-1)$	high dimensionality, convex, disconnected
ZDT4	10	$x_1 \in [0, 1],$ $x_i \in [-5, 5]^{n-1}$	$f_1(x) = x_1, f_2(x) = g(x)[1 - \sqrt{x_1/g(x)}]$ $g(x) = 1 + 10(n-1) + \sum_{i=2}^n [x_i^2 - 10 \cos(4\pi x_i)]$	non-convex, 21^9 local Pareto fronts
ZDT6	10	$[0, 1]^n$	$f_1(x) = 1 - \exp(-4x_1) \sin^6(6\pi x_1)$ $f_2(x) = g(x)[1 - (f_1(x)/g(x))^2]$ $g(x) = 1 + 9[(\sum_{i=2}^n x_i)/(n-1)]^{0.25}$	non-convex, non-uniformly spaced
ZDT1.1	10	$[0, 1]^n$	$f_1(x) = x_1, f_2(x) = g(x)[1 - \sqrt{x_1/g(x)}]$ $g(x) = 1 + 9(\sum_{i=2}^n (x_i - x_1)^2)/(n-1)$	convex, variable dependent
DTLZ1	12	$[0, 1]^n$	$f_1(x) = 0.5x_1x_2(1 + g(\vec{x}_M)), f_2(x) = 0.5x_1(1 - x_2)(1 + g(\vec{x}_M))$ $f_3(x) = 0.5(1 - x_1)(1 + g(\vec{x}_M))$ $g(\vec{x}_M) = 100[\vec{x}_M + \sum_{x_i \in x_M} (x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5))]$	3 objectives, hyper-plane
DTLZ7	22	$[0, 1]^n$	$f_1(x) = x_1, f_2(x) = x_2, f_3(x) = (1 + g(\vec{x}_M))h(f_1, f_2, g)$ $g(\vec{x}_M) = 1 + \frac{9}{ \vec{x}_M } \sum_{x_i \in x_M} x_i$ $h(f_1, f_2, g) = 3 - \sum_{i=1}^2 [\frac{f_i}{1+g} (1 + \sin(3\pi f_i))]$	3 objectives, disconnected

其中, d_i 表示 Q 中第 i 个个体与 P^* 之间在目标空间的最小欧氏距离. γ 值越小, 表明算法所得非劣解集 Q 越接近所求问题的真正 Pareto 前沿, 算法收敛性越好.

2) 多样性 Δ . 多样性用于评价算法的分布广度和均匀程度;

$$\Delta = (d_f + d_l + \sum_{i=1}^{|\mathcal{Q}|-1} |d_i - \bar{d}|) / (d_f + d_l + (|\mathcal{Q}|-1)\bar{d}) \quad (7)$$

其中, d_i 是非劣最优目标域 Q 中连续两个非劣解向量间的欧氏距离; \bar{d} 是所有 d_i 的均值; d_f 和 d_l 分别表示真正

Pareto 前沿近似集 P^* 中边界点和非劣解集 Q 边界点间的距离. Δ 值越小, 表明算法的多样性越好.

3.3 参数设置

对所有测试问题, 演化群体大小 $N=100$; 最大适应值评价次数 $Max_eval=10\ 000$; 缩放因子 $F=0.5$; 选择控制参数 $\lambda=0.1$; 正交设计指数 $J=2$. 对 ZDT1-4 和 ZDT6, DTLZ1, DTLZ7, 交叉概率 $CR=0.1$, 对 ZDT1.1, $CR=0.9$. 对 ZDT1-3, 正交设计水平数 $Q=29$; 对 ZDT4, ZDT6 和 ZDT1.1, DTLZ1, DTLZ7, $Q=21$; 为了使 Archive 群体中获得的非劣解个数大致为 100, 对 ZDT1-2, ZDT4 和 ZDT1.1, $\epsilon=[0.0075, 0.0075]^T$, 对 ZDT3, $\epsilon=[0.0025, 0.0025]^T$, 对 ZDT6, $\epsilon=[0.0060, 0.0070]^T$, 对 DTLZ1, $\epsilon=[0.02, 0.02, 0.05]^T$, 对 DTLZ7, $\epsilon=[0.05, 0.05, 0.05]^T$.

3.4 实验结果及分析

算法采用标准 C++ 语言实现, 每个问题采用不同的随

机种子独立运行 30 次, 统计平均值和方差. 测试环境为: CPU: P-IV 2.8G; 内存: 512M; 操作系统: Windows XP. 文中把 ϵ -ODEMO 的结果与 NSGA-II^[2], SPEA2^[4], MMDE-DE^[9] 和 MOEO^[27] 进行比较. 同时, 在其他参数设置与 ϵ -ODEMO 相同的情况下, 本文对 ϵ -DEMO (采用 ϵ 占优方法和随机初始化) 和 DEMO (采用随机初始化但不采用 ϵ 占优方法) 进行测试, 以验证正交初始化和 ϵ 占优方法的性能. 值得指出的是 NSGA-II^[2], SPEA2^[4], MMDE-DE^[9] 和 MOEO^[27] 的最大函数评价次数均为 25 000, 都远大于本文中算法的最大函数评价次数 10 000. 表 2 为各算法收敛性统计结果, 表 3 为多样性统计结果 (各算法第一行表示平均值, 第二行表示方差).

从表 2 可以看出, ϵ -ODEMO 具有很好的收敛性, 其收敛性对所有测试函数均优于所比较的四个算法. 且收敛性的方差很小, 表明 ϵ -ODEMO 具有很高的稳定性, 能在所有 30 次运算中均收敛于所优化问题的真正 Pareto 前沿. NSGA-II 和 SPEA2 的收敛性较差, 在 ZDT4, ZDT6 和 DTLZ1 上, 这两个算法容易陷入问题的局部 Pareto 前沿. DEMO 收敛性较差, 特别在 ZDT2 和 ZDT6 上其性能较差, 表明该算法在求解非凸 Pareto 前沿问题时性能很差. 在引入 ϵ 占优方法后, ϵ -DEMO 的收敛性相对于 DEMO, 具有一定的改进, 但与 ϵ -ODEMO 相比在所有测试问题上其收敛性都差很多. 这表明了 ϵ 占优方法和正交群体初始化可以有效增强算法的收敛性, 指导算法收敛于真正 Pareto 前沿. MMDD-DE 和

MOEO 算法只测试了 ZDTs 问题, 其收敛性较好, 但比 ϵ -ODEMO 的性能稍差.

Table 2 Comparison of the convergence metric τ . A result in Boldface indicates the better value obtained.

表 2 各算法收敛性比较(黑体部分表示算法所得到的较好解)

Algorithm	ZDT1	ZDT2	ZDT3	ZDT4	ZDT6	ZDT1.1	DTLZ1	DTLZ7
NAGS-II [2]	0.000894	0.000824	0.43411	3.22764	7.80680	0.00392	3.75997	0.03241
	0	0	0.000042	7.30763	0.001667	0.001976	1.89255	0.00457
SPEA2 [4]	0.023285	0.16762	0.018409	4.9271	0.23255	0.02572	3.235731	0.03898
	0	0.000815	0	2.703	0.004945	0.00245	1.38236	0.00365
MMDD-DE [9]	0.000968	0.000752	0.001196	0.001017	0.000598	NA	NA	NA
	0	0	0	0.000120	0.000032	NA	NA	NA
MOEO [27]	0.001277	0.001355	0.004385	0.008145	0.000630	NA	NA	NA
	0.000697	0.000897	0.001910	0.004011	0.000033	NA	NA	NA
DEMO	0.00554	0.14345	0.07951	0.00135	0.57403	0.0064	0.06361	0.07058
	0.000692	0.01237	0.00922	0.000710	0.02934	0.00493	0.11229	0.00904
ϵ -DEMO	0.00476	0.01682	0.0086	0.000938	0.54372	0.00232	0.040365	0.05346
	0.00518	0.00911	0.000807	0.000265	0.61489	0.000353	0.02364	0.00359
ϵ -ODEMO	0.000187	0.000534	0.000218	0.000187	0.000615	0.001880	0.00438	0.02036
	0.000012	0.000570	0.000011	0.000012	0.000014	0.000438	0.000250	0.00105

Table 3 Comparison of the diversity metric Δ . A result in Boldface indicates the better value obtained.

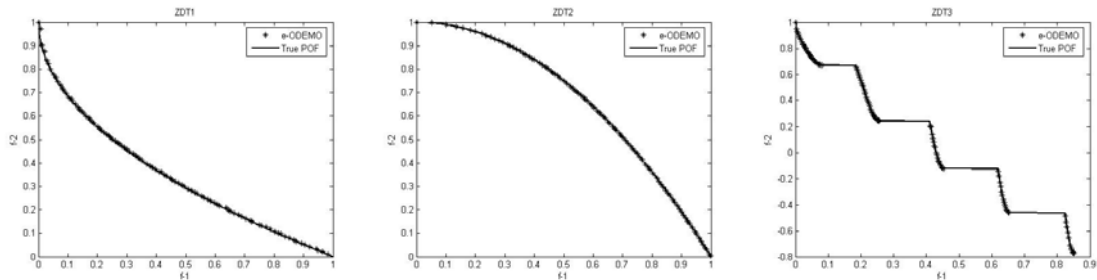
表 3 各算法多样性比较(黑体部分表示算法所得到的较好解)

Algorithm	ZDT1	ZDT2	ZDT3	ZDT4	ZDT6	ZDT1.1	DTLZ1	DTLZ7
NAGS-II [2]	0.463293	0.435112	0.575606	0.479475	0.644477	0.65028	0.95023	0.57615
	0.0414622	0.024607	0.005078	0.009841	0.035042	0.06389	0.2654	0.06149
SPEA2 [4]	0.154723	0.33945	0.4691	0.8239	1.04422	0.70987	0.8507665	0.37781
	0.0008738	0.001755	0.005265	0.002883	0.158106	0.0826	0.63254	0.05698
MMDD-DE [9]	0.253462	0.302548	0.308754	0.348541	0.421578	NA	NA	NA
	0.098544	0.003215	0.002143	0.002143	0.026842	NA	NA	NA
MOEO [27]	0.327140	0.285062	0.965236	0.275567	0.225468	NA	NA	NA
	0.065343	0.056978	0.046958	0.183704	0.033884	NA	NA	NA
DEMO	0.33648	0.68299	0.70669	0.3252	0.8625	0.68095	0.46383	0.50651
	0.03744	0.04095	0.05365	0.03349	0.04453	0.05608	0.07482	0.07238
ϵ -DEMO	0.32187	0.36593	0.48575	0.3187	0.51864	0.58715	0.43465	0.50235
	0.02262	0.0347	0.02748	0.01268	0.27761	0.057	0.08316	0.06324
ϵ -ODEMO	0.31656	0.27578	0.46081	0.31758	0.18415	0.36348	0.15784	0.33393
	0.01536	0.02396	0.02587	0.01015	0.01512	0.02714	0.01434	0.0503

表 3 表示各算法多样性统计结果. 从表中可以看出, ϵ -ODEMO 在 ZDT2, ZDT3, ZDT4, ZDT6, ZDT1.1, DTLZ1 和 DTLZ7 上优于所比较的算法. 在 ZDT1 上 SPEA2 的多样性最好, 但从表 2 可以看出其收敛性很差. 对 DEMO, ϵ -DEMO 和 ϵ -ODEMO 这三个算法, 从表 3 中可以得出与表 2 类似的结果, ϵ -ODEMO 的多样性最好, 其次是 ϵ -DEMO, 而 DEMO 的在所有测试问题上多样性最差. 这进一步验证了 ϵ 占优方

法和正交群体初始化对算法性能的提高. 与收敛性相似, MMDD-DE 和 MOEO 的收敛性稍差于 ϵ -ODEMO.

Zitzler 等^[25]指出仅采用数值评价标准不能完全反映算法性能的优劣, 为此, 本文绘制出 ϵ -ODEMO 所得的非劣解集, 以更直观地了解该算法的性能. 图 2 为 ϵ -ODEMO 一次运行所得非劣解集绘制的 Pareto 前沿(其中 True POE 表示所优化问题的真正 Pareto 前沿).



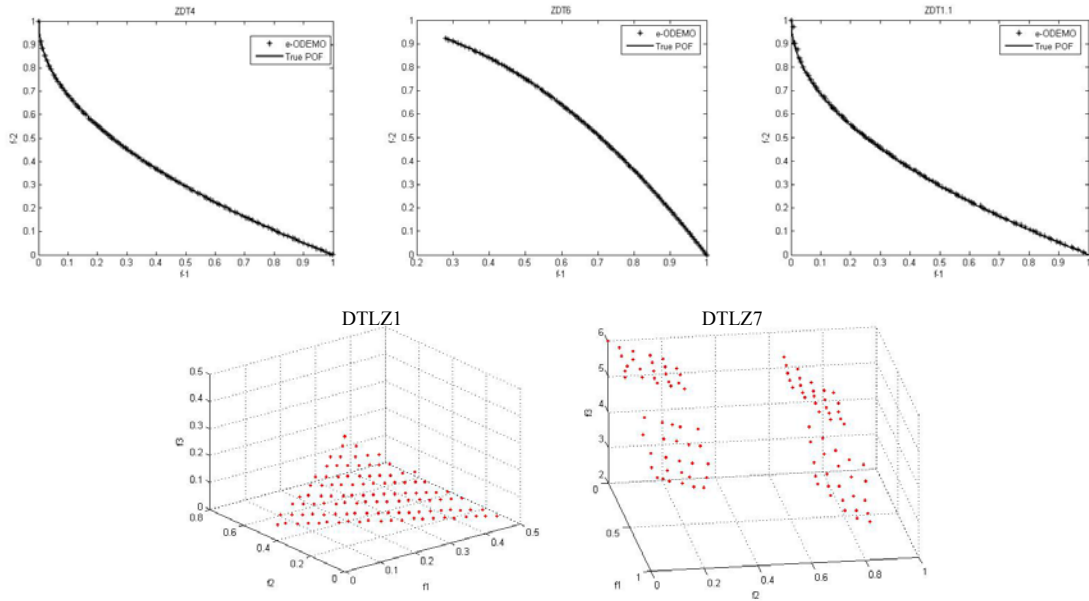


Fig. 2 Non-dominated solutions of final Archive obtained by ϵ -ODEMO on all test problems

图 2 ϵ -ODEMO 所得非劣解集的 Pareto 前沿

从图 2 可以看出, ϵ -ODEMO 的 Pareto 前沿极好地逼近了所求解问题的真正 Pareto 前沿, 完全覆盖了整个 Pareto 前沿且比较均匀.

3.5 CR 取值的影响

在 DE 算法中, 交叉概率 CR 的取值对算法的性能影响较大, 为此, 本研究中对 CR 取 0.1, 0.2, ..., 1.0 的 10 个不同

取值进行了实验研究, 算法对每个取值独立运行 30 次, 图 4 给出了 ϵ -ODEMO 在不同 CR 值下各测试函数平均收敛性和多样性变化曲线. (注: 不同 CR 值对 DTLZ1 和 DTLZ7 的影响与对变量无关的 ZDT 函数影响类似, 此处不再做单独讨论).

由图 4 可以看出, 对于变量无关的问题(ZDT1-4, ZDT6), 当 CR 的值大于 0.8 时, ϵ -ODEMO 的性能不断减低, 容易陷

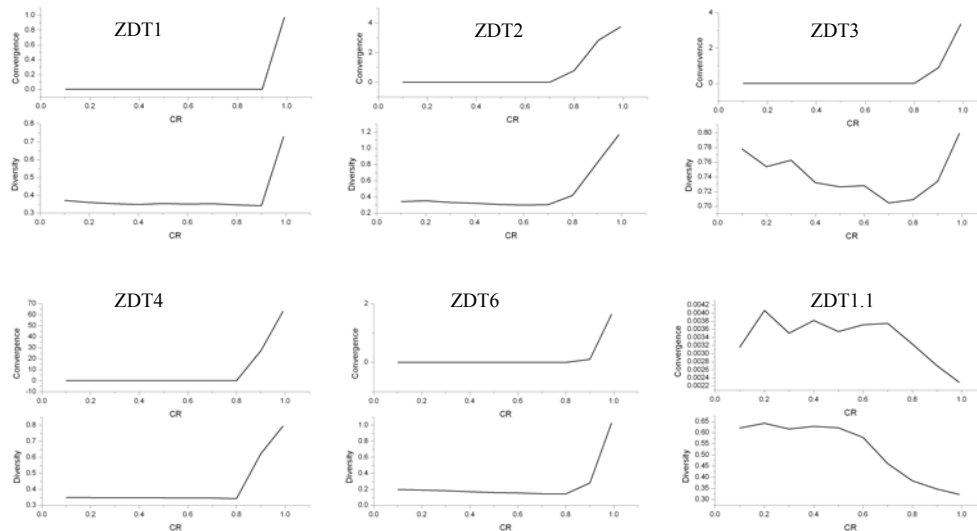


Fig. 4 Illustration of convergence and diversity with different CR value for test problems

图 4 在不同 CR 下各测试函数 Convergence 和 Diversity 评价标准变化曲线图

入局部 Pareto 前沿; 当 CR 的值较小时, 该算法的性能较好. 而对于变量相关的问题 ZDT1.1, ϵ -ODEMO 的性能随 CR 值的增大在提高, 当 CR 值在[0.8,1.0]之间取值时, 算法的性能较好. 其原因在于, 对于变量无关的问题, 当 CR 的值比较大时, DE 算子容易破坏原有较好的模式, 从而导致算法性能下降. 而对变量相关问题, 由于各变量之间存在相关性, 因此, 较大 CR 值可以更多利用 DE 算子的搜索信息, 探索

好的模式, 提高算法性能.

3.6 混合选择机制性能验证

为了进一步验证混合选择机制对算法性能的改进, 笔者在选择控制参数 λ 取 0.1, 0.2, ..., 1.0 这 10 个值时对所有测试函数进行测试, 其他参数与 3.3 节中设置一样, 每个函数运行 30 次. 值得提出的是当 $\lambda = 1.0$ 时, 表示算法只从演化群体中选择个体, 不利用 Archive 中个体的信息. 图 5 为

ϵ -ODEMO 在 ZDT1 上收敛性和多样性变化曲线; 图 6 为 ϵ -ODEMO 在 ZDT1.1 上收敛性和多样性变化曲线. 其中 X 轴表示 λ (1 对应 0.1, 2 对应 0.2, 以此类推); Y 轴表示收敛性

或多样性的值; 左图为收敛性曲线; 右图为多样性曲线.

从图 5 可以看出, 虽然 λ 取值对多样性的影响规律不明显, 但是, 当 λ 取值增大时, 收敛性总体上在不断变差. 在

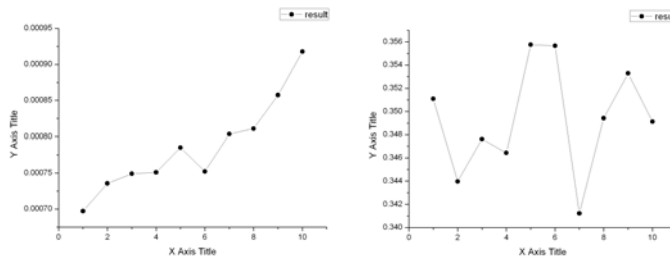


Fig. 5 Illustration of convergence and diversity with different λ value for ZDT1

图 5 不同 λ 取值下 ZDT1 函数 Convergence 和 Diversity 变化曲线

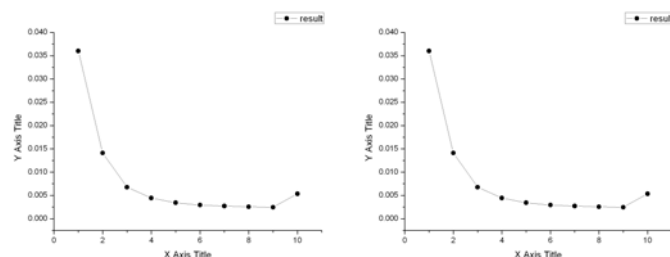


Fig. 6 Illustration of convergence and diversity with different λ value for ZDT1.1

图 6 不同 λ 取值下 ZDT1.1 函数 Convergence 和 Diversity 变化曲线

图 6 中, 除了 $\lambda=1.0$ 外, 其收敛性和多样性都随 λ 增大而得到改进. 由此可以得出, 对于变量无关问题, 选择控制参数 λ 应取较小的值, 因为算法能够较快得到较好的解, 并存储在 Archive 中, λ 取较小的值可使算法尽快利用 Archive 中的有效个体指导演化; 而对变量相关问题 λ 应取较大的值, 由于所优化问题变量间相互关联, 在演化初期 Archive 中存储的解不一定是有效解, 对演化的指导作用不大, 如果过多利用 Archive 中的个体信息有可能会使算法陷入局部 Pareto 前沿(如当 $\lambda=0.1$ 时, 算法的收敛性和多样性都很差). 但是, 当 $\lambda=1.0$ 时, 无论对变量无关问题还是变量相关问题, 其性能都较差, 这表明合理利用 Archive 中个体的信息可以有效指导演化, 提高算法性能.

4 总结及今后工作

本文提出了一种基于 ϵ 占优的正交多目标演化算法 ϵ -ODEMO. 新算法采用正交实验设计和连续空间量化的方法产生初始群体代替传统演化算法中的随机产生方法, 增强了算法的利用能力和有效性. ϵ -ODEMO 采用第二个群体 Archive 来保存得到的非劣解, 并利用 ϵ 占优方法更新 Archive 群体, 保证了算法的收敛性和多样性. 同时, 为了加快算法的收敛, 新算法利用基于随机选择和精英选择的混合选择机制, 增强了 Archive 中的个体对算法的指导作用.

通过 8 个标准测试函数对 ϵ -ODEMO 进行测试, 并与 NSGA-II, SPEA2, MMDD-DE, MOEO, DEMO 和 ϵ -DEMO 算法进行对比, 表明 ϵ -ODEMO 具有很好的收敛性和多样性, 且该算法的稳定性很好, 在 30 次运算中均能有效收敛于所

有测试问题的真正 Pareto 前沿. 而且, 通过实验对比, 验证了正交群体初始化, 混合选择机制, 和 ϵ 占优方法的有效性. 同时, 本文通过实验的方法初步讨论了 DE 算法中 CR 取值对算法性能的影响, 结论是对于变量无关的问题, CR 的取值应该较小; 而对于变量相关问题, CR 的取值较大. 并且讨论了选择控制参数 λ 取值对算法的影响, 结论与 CR 的取值类似: 对于变量无关的问题, λ 的取值应该较小; 而对于变量相关问题, λ 的取值较大. 今后的工作将致力于在算法中引进约束函数处理机制, 以增强算法的约束函数处理能力, 以便把算法应用到工程优化中.

Reference:

- [1] Carlos A. Coello Coello: Evolutionary Multi-objective Optimization: A Historical View of the Field[J], IEEE Computational Intelligence Magazine, 2006, 1(1): 28-36.
- [2] Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II[J]. IEEE Transactions on Evolutionary Computation, 2002, 6(2): 182-197.
- [3] T. Robič and B. Filipič: DEMO: Differential Evolution for Multiobjective Optimization[C]. The Third International Conference on Evolutionary Multi-Criterion Optimization, 2005, 520 - 533.
- [4] Zitzler, E., Laumanns, M., Thiele, L.: SPEA2: Improving the strength Pareto evolutionary algorithm. Technical Report 103, Computer Engineering and Networks Laboratory (TIK), 2001.
- [5] Knowles, J.D., Corne, D.W.: Approximating the Nondominated Front Using the Pareto Archived Evolution Strategy. Evolutionary Computation, 2000, 8(2): 149 - 172.

- [6] Kukkonen, S., Lampinen, J.: GDE3: the third evolution step of generalized differential evolution [C]. IEEE Congress on Evolutionary Computation, 2005, (1): 443- 450.
- [7] Zeng SY, Wei W, Kang LS, *et al.* A multi-objective evolutionary algorithm based on orthogonal design[J]. Chinese Journal of Computers, 2005, 28(7): 1153 - 1162 (in Chinese).
(曾三友, 魏巍, 康立山, 等. 基于正交设计的多目标演化算法[J]. 计算机学报, 2005, 28(7): 1153-1162.)
- [8] Shi C, Li QY, Shi ZZ. A Quick Multi-Objective Evolutionary Algorithm Based on Dominating Tree[J]. Journal of Software, 2007, 18(3): .505 - 516 (in Chinese).
(石川, 李清勇, 史忠植. 一种快速的基于占优树的多目标进化算法[J]. 软件学报, 2007, 18(3): 505 - 516.)
- [9] Zhang LB, Zhou CG, Ma M, *et al.* A Multi-Objective Differential Evolution Algorithm Based on Max-Min Distance Density [J]. Journal of Computer Research and Development, 2007, 44 (1): 177 - 184 (in Chinese).
(张利彪, 周春光, 马铭, 等. 基于极大极小距离密度的多目标微分进化算法[J]. 计算机研究与发展, 2007, 44 (1): 177 - 184.)
- [10] Cui XX, Lin C. A preference-based multi-objective concordance genetic algorithm[J]. Journal of Software, 2005, 16(5): 761 - 770
(崔逊学, 林闯. 一种基于偏好的多目标调和遗传算法[J]. 软件学报, 2005, 16(5): 761 - 770.)
- [11] Storn, R. and Price K.: Differential evolution--A simple and efficient heuristic for global optimization over continuous spaces[J]. Journal of Global Optimization, 1997, 11, 341 – 359.
- [12] Storn, R. and Price K.: Home Page of Differential Evolution. Available online at: <http://www.ICSI.Berkeley.edu/~storn/code.html>. 2003.
- [13] Antony W. Iorio and Xiaodong Li. Solving rotated multi-objective optimization problems using differential evolution[C]. In AI 2004: Advances in Artificial Intelligence, LNAI 3339, 2004, 861–872.
- [14] Hussein A. Abbass and Ruhul Sarker. The Pareto Differential Evolution Algorithm[J]. International Journal on Artificial Intelligence Tools, 2002, 11(4): 531–552.
- [15] S. Kukkonen and J. Lampinen. An Extension of Generalized Differential Evolution for Multi-objective Optimization with Constraints[C]. In Parallel Problem Solving from Nature - PPSN VIII, LNCS 3242, 2004, 752–761.
- [16] K. T. Fang and C. X. Ma: Orthogonal and Uniform Design[M]. Science Press, 2001 (in Chinese).
(方开泰, 马长兴. 正交与均匀实验设计[M]. 科学出版社, 2001.)
- [17] Y. W. Leung and Y. Wang: An Orthogonal Genetic Algorithm with Quantization for Global Numerical Optimization[J]. IEEE Transactions on Evolutionary Computation, 2001, 5(1): 41 – 53.
- [18] Gong W. Y., Cai Z. H., and Ling C. X. ODE: A Fast and Robust Differential Evolution Based on Orthogonal Design[C], 19th Australian Joint Conference on Artificial Intelligence, LNAI 4304, 2006, 12: 709-718.
- [19] Laumanns, M., Thiele, L., Deb, K., and Zitzler, E. Combining convergence and diversity in evolutionary multi-objective optimization[J]. Evolutionary Computation, 2002, 10(3): 263–282.
- [20] Deb, K., Mohan, M., Mishra, S. Evaluating the ϵ -Domination Based Multi-Objective Evolutionary Algorithm for a Quick Computation of Pareto-Optimal Solutions[J]. Evolutionary Computation, 2005, 13(4): 501-525.
- [21] Deb, K., Mohan, M., and Mishra, S. Towards a quick computation of well-spread Pareto-optimal solutions[C]. The Second Evolutionary Multi-Criterion Optimization, 2003, 222–236.
- [22] L.V. Santana-Quintero and Carlos A. Coello Coello. An Algorithm Based on Differential Evolution for Multi-Objective Problems[J], International Journal of Computational Intelligence Research, 2005, 1(2): 151-169
- [23] Zitzler, E., Deb, K., Thiele, L.: Comparison of multiobjective evolutionary algorithms: Empirical results [J]. Evolutionary Computation, 2000, 8, 173 – 195.
- [24] Zhou A.M., Zhang Q. F., Jin Y. C., *et al.* A model-based evolutionary algorithm for bi-objective optimization[C]. IEEE Congress on Evolutionary Computation, 2005, (3): 2568 - 2575.
- [25] Zitzler, E., Thiele, L., Laumanns, M., *et al.*: Performance Assessment of Multiobjective Optimizers: An Analysis and Review[J]. IEEE Transactions on Evolutionary Computation, 2003, 7(2): 117 – 132.
- [26] Deb, K., Thiele, L., Laumanns, M., and Zitzler, E., Scalable Test Problems for Evolutionary Multi-Objective Optimization. Zurich, Switzerland, Tech. Rep. 112, 2001.
- [27] Chen M.R., Lu Y.Z., A Novel Elitist Multiobjective Optimization Algorithm: Multiobjective Extremal Optimization[J], European Journal of Operational Research, 2008, 188(3): 637-651.



Gong Wenyin, born in 1979, Ph. D. candidate. His main research interests include Evolutionary Computation, Evolutionary Clustering Analysis, Multiobjective Optimization and its application in engineering design.

龚文引, 1979年生, 博士生, 主要研究方法为演化计算, 演化聚类分析, 多目标优化及其在工程设计中的应用。



Cai Zhihua, born in 1964, Professor and Ph. D. supervisor, senior membership of China Computer Federation. His main research interests include Data Mining, Evolutionary Computation, and Machine Learning.

蔡之华, 1964年生, 教授, 博士生导师, 计算机学会高级会员, 主要研究方向为数据挖掘, 演化计算和机器学习。

Research Background

In the last few years, evolutionary multi-objective optimization (EMO) has become a very popular topic from evolutionary computation community. However, to design an efficient and effective EMO algorithm to find the *near-optimal* and *near-complete* Pareto front is a challenging task. In this paper, we propose a novel differential evolution algorithm, which is based on the ϵ -dominance concept and orthogonal design method, to solve multi-objective optimization problems (MOPs) efficiently. The ϵ -domination method can obtain good Pareto set, and the orthogonal design method used in our approach can obtain good initial population, and hence, it can enhance the performance of the algorithm.

This research was supported by the Civil Space Project of the 11th Five-Year Plan under grant No. C5220061318, the Humanities Base Project of Hubei Province under grant No. 2004B0011, and the Natural Science Foundation of Hubei Province grant No. 2003ABA043.