

# 基于差异演化算法的动压滑动轴承多目标优化\*

卢青波<sup>1,2</sup> 张学良<sup>1</sup> 温淑花<sup>1</sup> 兰国生<sup>1</sup> 刘丽琴<sup>1</sup>

(1. 太原科技大学机械工程学院, 太原 030024; 2. 郑州职业技术学院, 郑州 450121)

**摘要:** 为研究流体动压轴承的多目标优化问题, 提出一种改进多目标差异演化算法。该算法在选择差分向量时, 对产生差分向量的两个个体比较其优劣, 用非支配解减去支配解, 引导个体向非劣解进化, 提高算法的收敛速度; 其次提出了种群修剪策略, 消除进化后期种群中相同个体引起的种群全局搜索能力下降的缺点, 以提升算法的全局寻优能力。通过与其它算法的比较, 发现该算法能有效避免“早熟”收敛, 具有较好的收敛速度和多样性。工程实例求解结果表明了算法的工程可行性。

**关键词:** 动压滑动轴承 多目标优化 差异演化算法

**中图分类号:** TP391.9; TH133.31 **文献标识码:** A **文章编号:** 1000-1298(2013)03-0230-07

## Multi-objective Optimization of Hydrodynamic Sliding Bearing Based on Differential Evolution Algorithm

Lu Qingbo<sup>1,2</sup> Zhang Xueliang<sup>1</sup> Wen Shuhua<sup>1</sup> Lan Guosheng<sup>1</sup> Liu Liqin<sup>1</sup>

(1. College of Mechanical Engineering, Taiyuan University of Science and Technology, Taiyuan 030024, China

2. Zhengzhou Technical College, Zhengzhou 450121, China)

**Abstract:** In order to solve multi-objective optimization problem of the hydrodynamic sliding bearing, a modified multi-objective differential evolution algorithm (MMODE) was proposed. The proposed algorithm provided a modified differential vector selection mechanism to improve the convergence speed and a population pruning strategy to maintain the population diversity. The vector selection mechanism compared two selected individuals and used the non-dominated individual minus the domination individual. Compared with several other evolutionary algorithms, the results showed that the proposed algorithm could overcome the premature convergence efficiently and had better convergence and diversity metrics. The results of engineering example showed the feasibility of the proposed algorithm.

**Key words:** Hydrodynamic sliding bearing Multi-objective optimization Differential evolution algorithm

### 引言

滑动轴承因其形式简单、承载能力强、回转精度高、润滑膜具有抗冲击作用等优点而被广泛应用于汽轮机、内燃机、轧钢机、大型电机及仪表上。流体动压滑动轴承是表面完全被油膜分开的一种滑动轴承, 虽然文献[1~4]从不同的角度采用不同方法对

流体动压轴承的优化设计做了一些工作, 但都采用了将多目标转化为单目标进行求解的方法, 并没有真正实现该问题的多目标优化设计。因此, 本文采用多目标差异演化算法来研究流体动压轴承的多目标优化问题。

差异演化算法(Differential evolution, DE)<sup>[5]</sup>是一种随机的并行直接全局搜索算法, 简单易用, 具有

收稿日期: 2012-03-29 修回日期: 2012-04-23

\* 高等学校博士学科点专项科研基金资助项目(20091415110002)、山西省自然科学基金资助项目(2008011027-1)、山西省研究生教育改革创新研究项目(20092016)和山西省研究生优秀创新项目(20093022)

作者简介: 卢青波, 博士生, 郑州职业技术学院讲师, 主要从事现代优化设计理论与方法研究, E-mail: jh0262@126.com

通讯作者: 张学良, 教授, 博士生导师, 主要从事进化算法、现代优化设计理论与方法研究, E-mail: zhang\_xue\_l@sina.com

稳健性和全局寻优能力,已在多个领域得到广泛应用<sup>[6]</sup>。目前,一些学者也把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异演化算法<sup>[7-13]</sup>。尽管目前对差异演化算法(差分进化算法)在多目标优化中的应用已经取得了很多成果,但是,设计高效、鲁棒的多目标差异演化算法仍是研究的热点之一。为了能有效求解多目标优化及增强多目标差异演化算法的鲁棒性,本文首先对差异演化算法进行改进,提出一种改进多目标差异演化算法(MMODE),并与其它改进算法及经典算法进行性能比较,最后应用该算法求解流体动压滑动轴承的多目标优化问题,获取该模型的Pareto曲线,并对其进行分析。

## 1 流体动压滑动轴承多目标优化模型

流体动压滑动轴承<sup>[2-3]</sup>的参数包括:轴承内径 $d$ 、轴承宽度 $b$ 、相对间隙 $\psi$ 、润滑油的动力粘度 $\eta$ 及轴颈转速 $n$ 等,一般情况下,轴承内径 $d$ 及轴颈转速 $n$ 都是已知的,其余参数待定。这里选取宽径比 $b/d$ 、相对间隙 $\psi$ 以及润滑油动力粘度 $\eta$ 作为设计变量,即设计变量为 $\mathbf{x} = (x_1, x_2, x_3) = (b/d, \psi, \eta)$ 。流体动压滑动轴承的性能可以从承载能力、传动效率及温升3方面考虑,分别用承载量系数 $C_F$ 、液体摩擦因数 $f$ 、发热量 $pv$ 来表征这些性能,其中 $p$ 为轴承的平均比压, $v$ 为轴颈的圆周速度,目标应使轴承承载能力强、传动效率高、温升低。

### (1) 承载量系数 $C_F$

轴承的承载能力可以用承载量系数 $C_F$ 来表征。 $C_F$ 越大,则轴承的承载能力越大。

又 $C_F = \frac{F\psi^2}{\eta\omega db}$ ,因此

$$\min f_1(\mathbf{x}) = \frac{1}{C_F} = \frac{\eta\omega d^2(b/d)}{F\psi^2} = \frac{\omega d^2 x_1 x_3}{F x_2^2} \quad (1)$$

式中  $\omega$ ——轴颈角速度,rad/s

$F$ ——轴承工作载荷,N

### (2) 液体摩擦因数 $f$

为使轴承的传动效率最大,必须使其阻力达到最小,即使液体摩擦因数最小。液体摩擦因数 $f =$

$\frac{\pi\eta\omega}{[F/(bd)]\psi} + 0.55\psi\xi$ ,其中 $\xi$ 是随轴宽径比而变化的系数。具体取值为: $b/d < 1$ 时, $\xi = (b/d)^{-1.5}$ ;  $b/d > 1$ 时, $\xi = 1.0$ 。

因此,目标函数为

$$\min f_2(\mathbf{x}) = f = \frac{\pi\eta\omega d^2(b/d)}{F\psi} + 0.55\psi\xi = \frac{\pi\omega d^2 x_1 x_3}{F x_2^2} + 0.55 x_2 \xi \quad (2)$$

### (3) 发热量函数

为使轴承发热量和磨损最小,应使轴承的 $pv$ 值最小,故目标函数为

$$\min f_3(\mathbf{x}) = pv = \frac{F}{bd}v = \frac{F}{(b/d)d^2}v = \frac{Fv}{x_1 d^2} \quad (3)$$

约束条件主要包括:

#### (1) 最小油膜厚度约束条件

为保证轴承形成液体摩擦润滑,应使轴承的最小油膜厚度 $h_{\min}$ 大于允许的最小油膜厚度 $[h_{\min}]$ ,文献<sup>[14]</sup>给出了 $h_{\min} = \frac{d^2\eta n}{18.36p\Delta C}$ 。将 $p = \frac{F}{bd}$ 、 $\Delta = \psi d/2$

及 $C = \frac{d+b}{b}$ 代入得 $h_{\min} = \frac{\eta n d^3 (b/d)^2}{9.18\psi (b/d+1)F}$ ,又 $[h_{\min}] = k(R_{z1} + R_{z2})$ ,其中 $k$ 为考虑几何形状误差、安装误差和轴颈变形等的安全系数,一般取2~3; $R_{z1}$ 、 $R_{z2}$ 分别为轴颈和轴承孔的表面粗糙度。因此约束条件为 $h_{\min} \geq [h_{\min}]$ ,即

$$k(R_{z1} + R_{z2}) - \frac{\eta n d^3 (b/d)^2}{9.18\psi (b/d+1)F} \leq 0 \quad (4)$$

#### (2) 比压约束条件

由 $p = \frac{F}{bd}$ 、 $p_{\min} \leq p \leq p_{\max}$ 得约束条件

$$p_{\min} \leq \frac{F}{(b/d)d^2} \leq p_{\max} \quad (5)$$

#### (3) 轴承宽径比约束条件

根据设计要求,轴承的宽径比应该满足

$$(b/d)_{\min} \leq b/d \leq (b/d)_{\max} \quad (6)$$

#### (4) 润滑油粘度约束条件

润滑油的粘度应满足的约束条件为

$$\eta_{\min} \leq \eta \leq \eta_{\max} \quad (7)$$

#### (5) 轴承相对间隙约束条件

设计时一般按经验公式确定 $\psi$ 的取值范围为

$$\psi_{\min} \leq \psi \leq \psi_{\max} \quad (8)$$

以上5项约束条件,其中第1项为条件约束,后4项均为边界约束。

因此其数学模型可以归结为

设计变量  $\mathbf{x} = (x_1, x_2, x_3) = (b/d, \psi, \eta)$

目标函数  $\min f_i(\mathbf{x}) \quad (i = 1, 2, 3)$

约束条件  $g(\mathbf{x}) \leq 0$

## 2 改进多目标差异演化算法

### 2.1 基本差异演化算法

差异演化算法是基于实数编码的进化算法,最初的群体是随机均匀产生的,每个个体看作 $D$ 维搜索空间中的一个实向量。令 $\mathbf{x}_i(g)$ 是第 $g$ 代的第 $i$ 个个体,则 $\mathbf{x}_i(g) = (x_{i1}(g), x_{i2}(g), \dots, x_{iD}(g))$ 。式中, $i \in [1, N]$ ;  $N$ 为种群规模; $g \in [1, t_{\max}]$ ;  $t_{\max}$ 为

最大进化代数;  $j \in [1, D]$ ; 且  $x_{ij}^L \leq x_{ij}(g) \leq x_{ij}^U, x_i^U, x_i^L$  为个体的上、下界。

其个体进化过程用数学语言表述为

$$v_{ij}(g+1) = \begin{cases} x_{pj} + \lambda(x_{pj} - x_{p_j}) & (\text{rand}(0,1) < R \text{ 或 } j = \text{rand}(D)) \\ x_{ij} & (\text{其他}) \end{cases} \quad (9)$$

$$x_i(g+1) = \begin{cases} v_i(g+1) & (f(v_i(g+1)) < f(x_i(g))) \\ x_i(g) & (f(v_i(g+1)) \geq f(x_i(g))) \end{cases} \quad (10)$$

式中,  $R$  为交叉概率, 一般在  $[0, 1]$  之间选择;  $\lambda$  为缩放因子, 在  $[0, 2]$  之间选择, 通常取 0.5;  $\text{rand}(0, 1)$  为  $[0, 1]$  上服从均匀分布的随机数;  $\text{rand}(D)$  为在  $[1, D]$  之间的随机整数;  $p_1, p_2, p_3$  为整数, 且  $p_1, p_2, p_3 \in [1, N], p_1 \neq p_2 \neq p_3 \neq i$ ;  $f(x)$  为评价函数。

2.2 外部存档及算法性能度量

在算法运行中, 应用外部档案来存储每一代所产生的非劣解, 随着迭代的进行, 外部档案的规模也逐渐增加, 其计算复杂度也增加, 当外部档案个数大于预设值 ACount 时, 采用拥挤距离方法计算每个非劣解的聚集距离, 然后对档案文件中的个体进行排序, 将超出 ACount 的非劣解从档案中删除。本文采用文献[9]提出的性能度量方法即收敛性能( $\gamma$ )和分散性能( $\zeta$ )进行性能度量。

2.2.1  $\gamma$  评价方法

$\gamma$  评价方法是评价所得解与 Pareto 最优面的趋近程度, 评价函数定义为

$$\gamma = \frac{\sum_{i=1}^{|Q|} d_i}{|Q|} \quad (11)$$

其中  $d_i = \min_{x_j^* \in P} \|f(x_i) - f(x_j^*)\|_2$

式中  $Q$ ——所得的非劣解集

$|Q|$ ——非劣解的个数

$x_j^*$ ——理论最优解

$f(x_i)$ ——第  $i$  个解在目标空间上的函数值

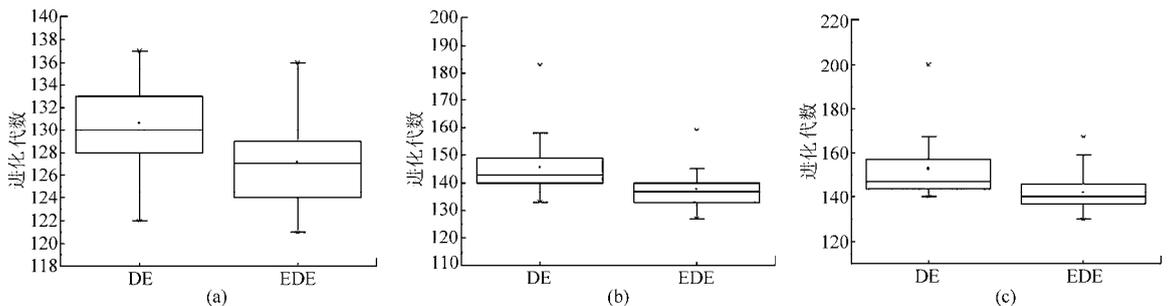


图 1 两种差分向量选择策略收敛代数比较

Fig. 1 Comparison of convergence generation with two differential vector strategies

(a) ZDT1 (b) ZDT2 (c) ZDT3

显然,  $\gamma$  越小, 说明与 Pareto 前沿越接近, 算法的收敛性能越好。

2.2.2  $\zeta$  评价方法

$\zeta$  评价方法是评价所得解的分布均匀性。评价函数的定义为

$$\zeta = \left[ \frac{1}{|Q| - 1} \sum_{i=1}^{|Q|-1} (\mu_i - \bar{\mu})^2 \right]^{\frac{1}{2}} \quad (12)$$

其中  $\mu_i = \min_{i \neq j} \|f(x_i) - f(x_j)\|_2, \bar{\mu} = \frac{\sum_{i=1}^{|Q|-1} \mu_i}{|Q| - 1}$

显然  $\zeta$  值越小, 说明解的分布更均匀, 多样性表现越好。

2.3 改进的差分向量选择策略

差分向量对于候选解的影响比较大, 同时也会影响到算法的收敛速度。为了使算法在现有条件下具有指导性, 加快算法的收敛速度, 本文提出以下选择策略。随机选择  $x_{p_2}$  和  $x_{p_3}$  两个向量作为差分向量的基向量: 若  $x_{p_2}$  优于  $x_{p_3}$ , 则差分向量按  $x_{p_2} - x_{p_3}$  运算; 若  $x_{p_3}$  优于  $x_{p_2}$ , 则差分向量按  $x_{p_3} - x_{p_2}$  运算; 若  $x_{p_2}$  与  $x_{p_3}$  等价, 则随机选择上面的一种进行运算。这样的操作不会增加算法的时间复杂度, 但是却能使每个个体的候选解朝着最优化的方向进化, 提升算法逼近最优解的能力, 加快算法的收敛速度。

设定种群大小为 100, 归档集最大容量为 50, 最大进化代数为 200,  $R$  为 0.1,  $\lambda$  为 0.5。算法终止条件为归档集达到最大容量且  $\gamma$  值小于等于 0.001, 对 ZDT1、ZDT2 及 ZDT3 测试函数各自独立运行 30 次, 图 1 给出了达到终止条件时的进化代数的统计盒图。其中, DE 算法为采用标准差异演化算法的差分向量选择策略, EDE 算法为采用改进的差分向量选择策略的改进差异演化算法。

从图 1 可以看出, 3 个测试函数中, 采用改进策略的算法 (EDE) 运行达到终止条件的代数的下四分位数、中位数和上四分位数都要小于 DE 算法的运行结果。这也表明, 改进策略能够提升算法的收敛速度。

### 2.4 种群修剪策略

种群的多样性会随着进化代数的增加而减小, 其中一个重要原因是, 在进化一段时间之后, 种群中会出现相同个体, 即位置相同的个体 (或适应度值完全相同个体), 这些个体的存在就会使原有的进化模式遭到破坏, 在 DE 算法中, 选择与当前个体不同的个体  $p_1$ 、 $p_2$ 、 $p_3$  产生候选解, 有了相同个体之后, 尽管  $p_1$ 、 $p_2$ 、 $p_3$  与当前个体不是同一个体, 但是其中某一个或者多个与当前个体是相同个体, 就会导致对当前个体的进化失败。种群修剪策略的伪代码如下:

```

Void PopPrun()
{
    S = NP - 1;
    for (i = 0; i < NP - 1; i++)
    {
        for(j = S; j > i; j--)
        {
            if (两者相等)
            {
                将当前比较个体 j 与当前最后一个个体 S 交换;
                S = S - 1;
                对第 S 个个体进行初始化操作;
            }
        }
    }
}
    
```

图 2 是归档集容量设置为 100, 种群大小设置为 100, 最大进化代数为 250, 群体中相同个体数随进化代数变化曲线。从图 2 可以看出, 群体中出现相同个体数目的最大值接近 25, 因此从 5 开始间隔 5 进行实验。表 1 是以 ZDT1 函数为研究对象, 种群中的相同个体数大于 S 时, 执行种群修剪策略所取得的统计结果比较。从表中数据可以看出, 在 S 取 20 时, 对 ZDT1 求解结果最好。即 S = 20 时, 不仅起

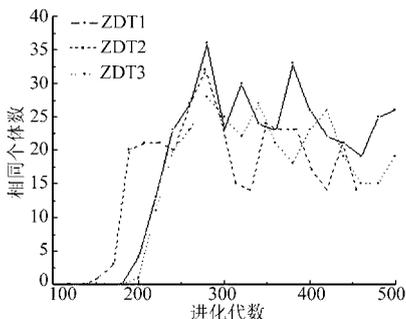


图 2 相同个体随代数变化曲线

Fig. 2 Curves of same individual with generation

到了维持种群多样性的目的, 也起到了提升算法性能的作用。为降低种群修剪策略的执行次数, 本文在最大进化代数执行一半之后, 检查种群中相同个体的数目, 当其大于预先设定的数值时, 执行种群修剪策略。

表 1 不同 S 时 ZDT1 函数统计结果比较

Tab. 1 Comparison of statistical results of ZDT1 by different S

S	$\gamma$		$\zeta$	
	平均值	方差	平均值	方差
5	0.002 23	$9.03 \times 10^{-4}$	$3.02 \times 10^{-4}$	$1.48 \times 10^{-4}$
10	0.002 04	$1.11 \times 10^{-3}$	$2.91 \times 10^{-4}$	$1.38 \times 10^{-4}$
15	0.001 78	$6.59 \times 10^{-4}$	$2.65 \times 10^{-4}$	$1.19 \times 10^{-4}$
20	0.001 64	$8.09 \times 10^{-4}$	$2.54 \times 10^{-4}$	$9.13 \times 10^{-5}$
25	0.001 78	$8.65 \times 10^{-4}$	$2.57 \times 10^{-4}$	$9.34 \times 10^{-5}$

### 2.5 MMODE 算法流程

- (1) 初始化种群及算法参数。
- (2) 初始化档案文件 A。

(3) 对每个个体执行以下步骤: ①随机选择 3 个个体  $p_1$ 、 $p_2$ 、 $p_3$ , 判断  $x_{p_2}$  与  $x_{p_3}$  的优劣关系, 按 2.3 节的改进差分向量方法执行式 (9) 的操作。②执行式 (10) 的选择操作。③用新产生的个体更新归档集。若新个体不受归档集个体的支配, 则将其插入 A 中, 删除 A 中所有受新个体支配的解; 若新个体与归档集中的个体相互支配且归档集未达到最大限度, 则将其插入 A 中。若归档集 A 已满, 则使用聚集距离方法对其消减。

(4) 判断是否达到执行 2.4 节种群修剪策略的条件, 达到, 则执行; 否则, 转步骤 (5)。

(5) 进化代数加 1, 判断是否达到收敛条件, 若未达到, 则转步骤 (3); 否则, 输出归档集, 获得 Pareto 最优解集。

## 3 数值仿真

### 3.1 测试函数及算法参数设置

采用测试函数 ZDT1、ZDT2、ZDT3、ZDT4、ZDT6<sup>[15]</sup> 对算法进行测试。

在种群修剪策略中, 由于种群中有新个体的加入, 需要对新产生的个体进行函数评价, 因此, 采用最大次数作为终止条件, 设置为 25 000 次。为减少空间复杂度, 设置 MMODE 算法的种群大小  $N = 50$ 。交叉概率  $R = 0.1$ , 归档集的最大容量为 100。设置 S 为 20, 为保持算法参数的个数不变, 将缩放因子 F 设置为 (0, 1) 之间的随机数。算法采用标准 C++ 语言实现, 每个测试函数独立运行 30 次, 统计平均值与方差。所得结果与 NSGA-II<sup>[9]</sup>、SPEA2<sup>[16]</sup>、

DEMO<sup>[8]</sup>、MMDE-DE<sup>[12]</sup>进行比较。表2为各算法的收敛性统计结果,表3为多样性统计结果(各算法第1行表示平均值,第2行表示方差)。

### 3.2 实验结果及分析

从表2数据可以看出,MMODE在求解ZDT1、ZDT2、ZDT3、ZDT4时,其收敛性指标均优于其余算法。而在求解ZDT6时,MMDE-DE算法的收敛性最好,其次是MMODE算法。从方差结果可以看出,MMDE-DE算法的稳定性最好,MMODE算法的稳定性不如MMDE-DE算法。这表明一方面改进的差分

向量选择策略促进了算法的快速收敛;另一方面,种群修剪策略也对种群的进化起到了作用,提升了算法的全局收敛性能,但同时也使算法的稳定性有所下降。

从表3数据可以看出,MMODE在ZDT1、ZDT2、ZDT3及ZDT4上的多样性指标都优于其余算法。而在求解ZDT6时,MMDE-DE算法的多样性最好,其次是DEMO算法,MMODE算法只是优于NSGA-II和SPEA2算法。从方差结果可以看出,本文算法能够稳定地保证解的分布性。

表2 各算法收敛性比较

Tab.2 Comparison of convergence metric

算法	ZDT1	ZDT2	ZDT3	ZDT4	ZDT6
NSGA-II	0.000 894	0.000 824	0.434 11	3.227 64	7.806 80
	0	0	0.000 042	7.307 63	0.001 667
SPEA2	0.023 285	0.167 62	0.018 409	4.927 1	0.232 55
	0	0.000 815	0	2.703	0.004 945
DEMO	0.001 083	0.000 75	0.001 178	0.001 037	0.000 629
	0.000 113	0.000 045	0.000 059	0.000 134	0.000 044
MMDE-DE	0.000 968	0.000 752	0.001 196	0.001 017	0.000 598
	0	0	0	0.000 120	0.000 032
MMODE	0.000 083	0.000 077	0.000 521	0.007 55	0.012 04
	0.000 087	0.000 061	0.000 438	0.027 12	0.023 04

表3 各算法多样性比较

Tab.3 Comparison of diversity metric

算法	ZDT1	ZDT2	ZDT3	ZDT4	ZDT6
NSGA-II	0.463 293	0.435 112	0.575 606	0.479 475	0.644 477
	0.041 462 2	0.024 607	0.005 078	0.009 841	0.035 042
SPEA2	0.154 723	0.339 45	0.469 1	0.823 9	1.044 22
	0.000 873 8	0.001 755	0.005 265	0.002 883	0.158 106
DEMO	0.325 237	0.329 151	0.309 436	0.359 905	0.442 308
	0.030 249	0.032 408	0.018 603	0.037 672	0.039 255
MMDE-DE	0.253 462	0.302 548	0.308 754	0.348 541	0.421 578
	0.098 544	0.003 215	0.002 143	0.002 143	0.026 842
MMODE	0.000 22	0.000 19	0.005 22	0.035 67	0.626 48
	0.000 65	0.000 05	0.000 44	0.064 56	0.128 31

为了能直观地了解算法的性能,绘制了MMODE算法一次运行所得非劣解集的Pareto前沿,如图3所示。其中True Pareto Front表示所优化问题的真实Pareto前沿。从图中可以看出,MMODE的Pareto前沿极好地逼近了所求问题的真实Pareto前沿,且完全覆盖了整个Pareto前沿,尤其是求得了各个目标函数的边界值,而且分布比较均匀。

### 3.3 高维多目标函数测试

由于本文为3个目标函数的问题,为测试本文算法在高维多目标下的性能,采用DTLZ1、

DTLZ2<sup>[17]</sup>进行测试,算法的结果如图4所示。实验时,最大进化代数设置为600代,其余参数设置参照3.1节。

图4得到的非劣解很接近理想值,并且分布也很均匀,表明本文算法的收敛性较好,并且由于引入了高斯变异,也提高了算法克服局部最优能力,获取全局最优解的能力。

## 4 工程实例

设计某齿轮箱的流体动压润滑径向滑动轴承,已

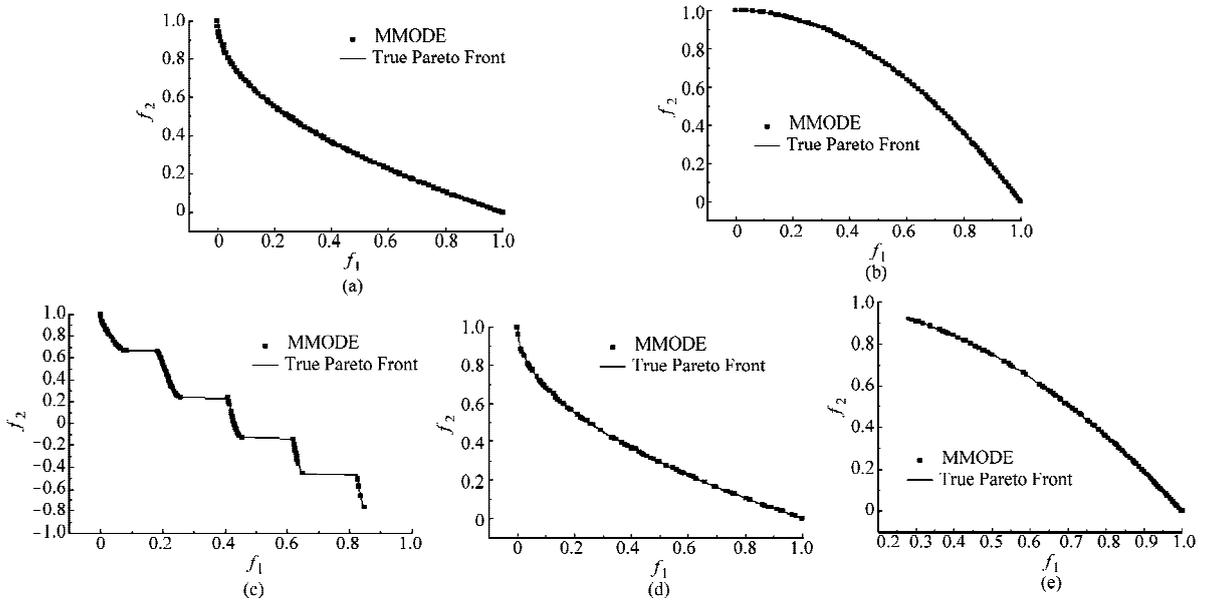


图3 MMODE算法与真实 Pareto 前沿比较

Fig.3 Comparison of MMODE algorithm and True Pareto Front

(a) ZDT1 (b) ZDT2 (c) ZDT3 (d) ZDT4 (e) ZDT5

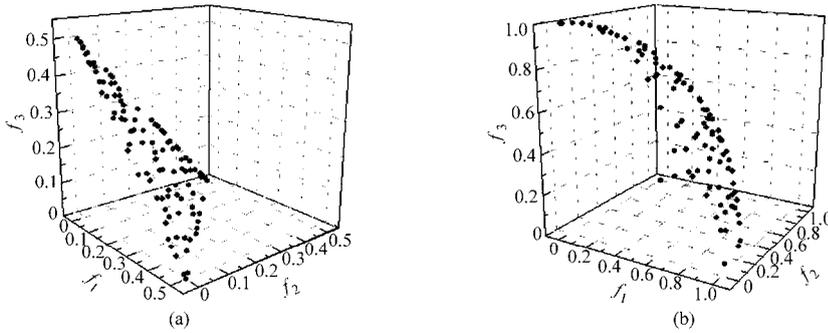


图4 MMODE 获得的 Pareto 曲线

Fig.4 Pareto curves by MMODE

(a) DTLZ1 (b) DTLZ2

知工作载荷为 35 000 N,轴的直径  $d = 100$  mm,轴的转速为  $n = 1\ 000$  r/min,轴承为剖分式。优化设计模型为

$$F = (f_1(\mathbf{x}), f_2(\mathbf{x}), f_3(\mathbf{x}))$$

$$\mathbf{x} = (x_1, x_2, x_3) = (b/d, \psi, \eta)$$

$$\min f_1(\mathbf{x}) = 2.9905 \times 10^{-5} x_1 x_2^{-2} x_3$$

$$\min f_2(\mathbf{x}) = 9.3901 \times 10^{-5} x_1 x_2^{-2} x_3 + 0.55 x_2 \varphi(x_1)^{-1.5}$$

$$\min f_3(\mathbf{x}) = 18.317 x_1$$

$$\varphi(x_1) = \begin{cases} 1 & (x_1 \geq 1) \\ x_1 & (x_1 < 1) \end{cases}$$

$$\text{s.t. } g(\mathbf{x}) = 9.6 \times 10^{-6} -$$

$$2.915 \times 10^{-5} x_1^2 x_3 / [x_2(x_1 + 1)] \leq 0$$

$$0.5 \leq x_1 \leq 1.2$$

$$0.001 \leq x_2 \leq 0.002$$

$$0.02 \leq x_3 \leq 0.04$$

该优化问题是一个 3 目标函数的约束优化问题,本文采用如下方法<sup>[13]</sup>处理约束。两个候选解如

果都在可行域内,则 Pareto 占优的候选解保留;两个候选解都不在可行域内,则对约束违背量小的保留;两个候选解,一个在可行域内,一个不在可行域内,则保留可行域内的候选解。

针对这样的约束处理方式,对群体进行初始化时,只要保证至少一个候选解在可行域内就可以,避免了在可行域内对个体进行初始化时反复初始化的缺点。另外也没有增加类似于罚函数法中的惩罚因子这样的多余参数,且易编程实现。

设置算法的参数为种群大小  $N = 50$ ,交叉概率  $R = 0.1$ ,归档集的最大容量为 100。设置  $S$  为 20,缩放因子  $\lambda$  设置为  $(0, 1)$  之间的随机数,最大进化代数设置为 1 000。图 5 给出了该问题的 Pareto 曲线。为了能更清晰地看出各个目标之间的关系,图 6 给出了承载量系数  $(1/f_1(\mathbf{x}))$  与液体摩擦因数  $f_2(\mathbf{x})$  之间的关系曲线,图 7 给出了承载量系数与发热量  $f_3(\mathbf{x})$  之间的关系曲线。

从图6可以看出,承载力系数与液体摩擦因数之间的关系为:当承载力系数小于6.8时,液体摩擦因数随着承载力系数的增大而减小;而当承载力系数大于6.8后,液体摩擦因数随着承载力系数的增大而增大。从图7可以看出,承载力系数与发热量函数几乎成正比关系。因此,在决策时,可以先从承载力系数与液体摩擦因数两个因素进行考虑,然后再考虑发热量的问题。甚至在散热条件较好情况下,可以忽略发热量函数的影响。

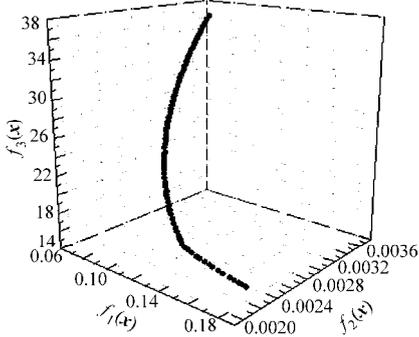
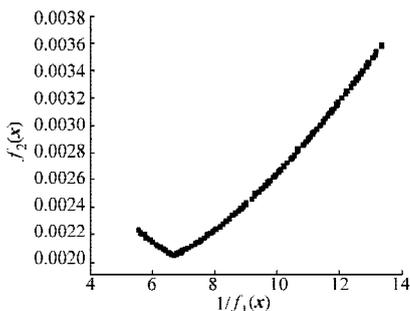
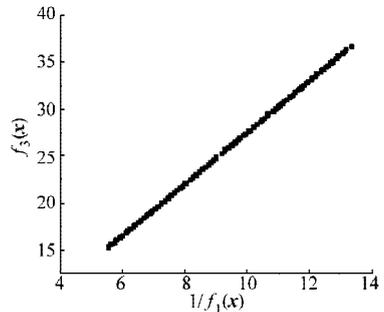


图5 Pareto 曲线

Fig.5 Pareto curves

图6  $C_F(1/f_1(x))$ 与 $f_2(x)$ 关系曲线Fig.6 Relation curves of  $C_F$  and  $f_2(x)$ 图7  $C_F(1/f_1(x))$ 与 $f_3(x)$ 关系曲线Fig.7 Relation curves of  $C_F$  and  $f_3(x)$ 

## 5 结束语

提出了一种改进多目标差异演化算法(MMODE)。该算法首先在选择差分向量时,对产生差分向量的两个个体比较其优劣,用非支配解减去支配解,引导个体向非劣解进化,提高算法的收敛速度;其次提出了种群修剪策略,消除进化后期,种群中相同个体引起的种群全局搜索能力下降的缺点,以提升算法的全局寻优能力。实验结果表明MMODE算法可以有效逼近真实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。通过5个测试函数对MMODE算法进行测试,并与NSGA-II、SPEA2、DEMO、MMDE-DE进行了性能比较,结果表明MMODE算法具有很好的收敛性和多样性,但其算法在稳定性上要比MMDE-DE算法差一些,下一步将致力于研究该算法的稳定性。两个高维函数的测试结果表明本文算法对于高维问题的表现也较好。工程实例的求解结果表明了本文算法的实用性及可行性。

## 参 考 文 献

- 魏锋涛,宋俐,李言. 最小偏差法在机械多目标优化设计中的应用[J]. 工程图学学报, 2011(3): 100~104.  
Wei Fengtao, Song Li, Li Yan. Application of minimum-deviation method to mechanical multi-objective optimal design[J]. Journal of Engineering Graphics, 2011(3): 100~104. (in Chinese)
- 张鄂,蒙娟,贾焕如. 液体动压径向滑动轴承的概率多目标优化设计[J]. 机械科学与技术, 2001, 20(2): 224~226.  
Zhang E, Meng Juan, Jia Huanru. The multi-aim probability optimal design of hydrodynamical journal bearing[J]. Mechanical Science and Technology, 2001, 20(2): 224~226. (in Chinese)
- 王增胜,刘保国,吴磊,等. 齿轮箱流体动压滑动轴承的多目标优化[J]. 机械传动, 2007, 31(5): 74~75.  
Wang Zengsheng, Liu Baoguo, Wu Lei, et al. Multi-objective optimization of hydrodynamic sliding bearing used in gear box[J]. Journal of Mechanical Transmission, 2007, 31(5): 74~75. (in Chinese)
- 张毅. 基于Matlab的汽轮机流体动压滑动轴承的优化设计[J]. 轴承, 2009(1): 4~7.  
Zhang Yi. Optimization design of hydrodynamic sliding bearing used in steam turbine based on Matlab[J]. Bearing, 2009(1): 4~7. (in Chinese)
- Rainer S, Price K. Differential evolution: a simple and efficient heuristic for global optimization over continuous spaces [J]. Journal of Global Optimization, 1997, 11(4): 341~359.
- 刘波,王凌,金以慧. 差分进化算法研究进展[J]. 控制与决策, 2007, 22(7): 721~729.  
Liu Bo, Wang Ling, Jin Yihui. Advances in differential evolution[J]. Control and Decision, 2007, 22(7): 721~729. (in Chinese)

- 19 Navarro O, Wu C J, Angeles J. Size-minimization of planar cam mechanisms[J]. Mechanism and Machine Theory, 2001, 36(3): 371 ~ 386.
- 20 石永刚, 吴央芳. 凸轮机构设计与应用创新[M]. 北京: 机械工业出版社, 2007.
- 21 Ji Z, Manna Y A. Size minimization of disc cams with roller-followers under pressure angle constraint[J]. Proc IMech E, Part C: Journal of Mechanical Engineering Science, 2008, 222(12): 2 475 ~ 2 484. (in Chinese)
- 22 常勇, 杨富富. 作平面运动滚子从动件盘形凸轮机构的第II类机构综合问题[J]. 机械工程学报, 2010, 46(21): 35 ~ 41. Chang Yong, Yang Fufu. Second mechanism synthesis task of disc cam mechanisms with roller follower moving in planar general motion[J]. Journal of Mechanical Engineering, 2010, 46(21): 35 ~ 41. (in Chinese)
- 23 常勇, 杨富富. 作平面运动滚子从动件形锁合凸轮机构的第II类机构综合问题[J]. 机械工程学报, 2012, 48(1): 39 ~ 46. Chang Yong, Yang Fufu. Research on second mechanisms synthesis task of positive-drive disc cam mechanisms with roller follower moving in general planar motion[J]. Journal of Mechanical Engineering, 2012, 48(1): 39 ~ 46. (in Chinese)
- 24 常勇, 李延平, 刘国祥. 按许用压力角设计最小尺寸的作平面复杂运动滚子从动件平面凸轮机构的解析法[J]. 机械工程学报, 1991, 27(4): 37 ~ 41. Chang Yong, Li Yanping, Liu Guoxiang. The analytics for designing minimum size disc cam mechanisms whose roller follower moving in general planar motion according to allowable pressure angle[J]. Chinese Journal of Mechanical Engineering, 1991, 27(4): 37 ~ 41. (in Chinese)
- 25 常勇, 杨富富, 胡志超, 等. 作平面运动滚子从动件盘形凸轮机构的广义第II类综合问题[J]. 机械工程学报, 2012, 48(15): 47 ~ 57. Chang Yong, Yang Fufu, Hu Zhichao, et al. Research on the general second mechanism synthesis task of disc cam mechanisms with roller follower moving in planar general motion[J]. Journal of Mechanical Engineering, 2012, 48(15): 47 ~ 57. (in Chinese)

~~~~~  
(上接第 236 页)

- 7 牛大鹏, 王福利, 何大阔, 等. 多目标混沌差分进化算法 [J]. 控制与决策, 2009, 24(3): 361 ~ 364. Niu Dapeng, Wang Fuli, He Dakuo, et al. Chaotic differential evolution for multiobjective optimization[J]. Control and Decision, 2009, 24(3): 361 ~ 364. (in Chinese)
- 8 Robic T, Filipic B. DEMO: differential evolution for multiobjective optimization [C] // Proc. of EMO'05, Berlin: Springer, 2005: 520 ~ 533.
- 9 Deb K, Pratap K, Agarwal S, et al. A fast and elitist multiobjective genetic algorithm: NSGA-II [J]. IEEE Transactions on Evolutionary Computation, 2002, 6(2): 182 ~ 197.
- 10 龚文引, 蔡之华. 基于 E 占优的正交多目标差分演化算法研究 [J]. 计算机研究与发展, 2009, 46(4): 655 ~ 666. Gong Wenyin, Cai Zhihua. Research on an E-domination based orthogonal differential evolution algorithm for multi-objective optimization[J]. Journal of Computer Research and Development, 2009, 46(4): 655 ~ 666. (in Chinese)
- 11 Abbass H, Sarker R. The Pareto differential evolution algorithm [J]. International Journal on Artificial Intelligence Tools, 2002, 11(4): 531 ~ 552.
- 12 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. Zhang Libiao, Zhou Chunguang, Ma Ming, 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)
- 13 吴亮红, 王耀南, 袁小芳, 等. 多目标优化问题的差分进化算法研究 [J]. 湖南大学学报: 自然科学版, 2009, 36(2): 53 ~ 57. Wu Lianghong, Wang Yaonan, Yuan Xiaofang, et al. Research on differential evolution algorithm for MOPS[J]. Journal of Hunan University: Natural Science, 2009, 36(2): 53 ~ 57. (in Chinese)
- 14 朱明道. 油膜厚度与滑动轴承精度的讨论[J]. 上海电力学院学报, 1998, 14(3): 25 ~ 29. Zhu Mingdao. Discussion on lubricant film thickness and bearing accuracy in the sliding bearing[J]. Journal of Shanghai Institute of Electric Power, 1998, 14(3): 25 ~ 29. (in Chinese)
- 15 Zitzler E, Deb K, Thiele L. Comparison of multiobjective evolutionary algorithms: empirical results [J]. Evolutionary Computation, 2000, 8(2): 173 ~ 195.
- 16 Zitzler E, Laumanns M, Thiele L. SPEA2: improving the strength Pareto evolutionary algorithm [C] // Proceedings of International Conference on Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems, Berlin: Springer, 2002: 95 ~ 100.
- 17 Deb K, Thiele L, Laumanns M, et al. Scalable test problems for evolutionary multi-objective optimization[R]. Swiss Federal Institute of Technology, Tech. Rep.: 112, 2001.