https://doi.org/10.1051/jnwpu/20193720232

基于 α-stable 分布的多目标粒子群算法研究及应用

樊华羽, 詹浩, 程诗信, 米百刚

(西北工业大学 航空学院, 陕西 西安 710072)

摘 要:多目标的粒子群算法(MOPSO)在各个领域的优化设计中得到了广泛应用及改进,但是目前仍然存在着在进化后期容易陷入局部最优导致收敛精度低、解的多样性差等问题。引入 α-stable 分布理论,发展建立了一种新的基于 α-stable 动态变异的多目标粒子群优化算法(ASMOPSO)。通过 α-stable 分布生成随机数对 PSO 算法的种群进行变异操作,增加种群的多样性,在算法中动态调整稳定性系数 α 实现变异范围和幅度的变化,从而使得改进的 ASMOPSO 算法具有兼顾计算精度和全局寻优的能力。使用 ZDT 系列无约束函数和带约束的 Tanaka 及 Srinivas 函数对改进前后的算法进行了测试,结果显示出了 ASMOPSO 算法的快速全局寻优性能。将改进后的算法应用到 RAE2822 跨音速翼型的减阻和力矩绝对值不增大的综合优化中,得到了较好的多目标气动优化结果。

关 键 词:多目标粒子群优化算法;α-stable 分布;动态变异;翼型设计;气动优化
中图分类号:V211 **文献标志码**:A **文章编号**:1000-2758(2019)02-0232-10

随着计算流体力学技术(CFD)和计算机软硬件的快速发展,各类数值优化算法结合先进 CFD 求解器进行高效的气动力优化设计成为研究的热点[1-2]。经过几十年的发展,在气动优化设计领域先后开发出了基于遗传算法[3]、粒子群算法[4]和差分进化[5]等智能优化算法的设计模式,对包括二维翼型、三维机翼、全机构型以及飞发一体化等问题进行了有效的气动优化设计研究,取得了丰硕的研究成果[6-7],其中应用最普遍的优化算法之一就是粒子群算法(PSO)。

粒子群算法通过模拟鸟群的觅食行为,将解空间中的设计变量当成一群质量和体积均为 0 的“鸟”,也就是所谓的粒子,将所求解问题的解当成鸟群寻觅的“食物”,以单个粒子的最优解和整个群体的最优解来引导下一代种群的搜索,以此来趋向最优解。该算法属于进化算法的一种,与遗传算法类似,也是从随机解出发,通过迭代寻找最优解,使用适应度来评价解的品质。相比遗传算法,粒子群算法规则简单,没有“交叉”和“变异”操作,仅是通过追随当前搜索到的最优值来寻找全局最优,容易实现,精度高、收敛快,但是该算法的缺点在于进化后期的早熟现象导致算法容易陷入局部最优。为此,研究人员先后建立了多个改进算法来提高其全局寻优能力。李丁等[8]将粒子群算法和差分进化算法结合,使得粒子群和差分群分层交换,共享最优信息,降低了算法陷入局部最优的风险,改进的算法在对 RAE288 翼型的减阻优化中取得了良好的效果;陈进等[9]提出一种反双曲余弦函数形式的非线性惯性权重表示法,改进了标准的粒子群算法,并对不同厚度的风力机翼型进行了优化,提高了相应的气动特性;李鑫等[10]采用基于二阶振荡和自然选择的改进粒子群算法对二维 SC-0717 超临界翼型进行了优化设计,显著提高了升阻比并降低了气动噪声。

尽管改进型算法改善了原算法容易陷入局部最优的问题,但是多数研究仍然局限在单点单目标状态,而气动优化设计是一种高维强非线性问题,考虑多个优化目标才能更好发挥设计的作用。为此,本文针对传统的粒子群算法进行改进,提出了一种基

收稿日期:2018-04-04

作者简介:樊华羽(1985—),西北工业大学博士研究生,主要从事飞行器气动及隐身多目标优化研究。

于 α-stable 分布的多目标粒子群算法。该算法将 α-stable 分布理论引入传统的粒子群算法,旨在改善粒子群智能算法的全局/局部搜索能力平衡问题。使用多个基准多目标函数对改进的多目标粒子群算法的性能进行了测试,并将改进后的多目标优化算法应用到 RAE2822 的气动优化设计中,对比分析了改进前后的算法在考虑减阻和力矩绝对值不减小 2 个目标下的优化效果。

1　基础知识

1.1　多目标优化问题

本文对于一个 n 维搜索空间 $S \in \mathbf{R}^n$,假定 $f_i(\boldsymbol{x})$, $i = 1,\cdots,m$ 是定义于 S 的目标数量为 m 的多目标优化向量,那么带约束的最小化连续多目标优化问题(muliti-objective optimization, MO)的通用数学模型可以定义如下

$$\begin{aligned} &\min \boldsymbol{f}(\boldsymbol{x}) = [f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \cdots, f_m(\boldsymbol{x})]^{\mathrm{T}} \\ &\text{s.t. } g_j(\boldsymbol{x}) \leqslant 0, \quad j = 1,\cdots,l \\ &h_k(\boldsymbol{x}) = 0, \quad k = 1,\cdots,p \end{aligned} \tag{1}$$

式中,$\boldsymbol{x} = [x_1, x_2, \cdots, x_n]$ 是决策变量向量,变量个数为 n,l 和 p 分别是不等式约束和等式约束的数量,$g_j(\boldsymbol{x})$ 是第 j 个不等式约束,$h_k(\boldsymbol{x})$ 是第 k 个等式约束。目标函数就是将决策空间映射到目标空间,即 $S \rightarrow \Omega \in \mathbf{R}^m$。

下面给出多目标优化问题中常用的 Pareto 最优相关定义。

定义 1　对于任意 2 个向量 $\boldsymbol{u} = (u_1, \cdots, u_m)$ 和 $\boldsymbol{v} = (v_1, \cdots, v_m)$,当且仅当:$\forall i = 1,2,\cdots,m, u_i \leqslant v_i \wedge \exists j = 1,2,\cdots,m, u_j < v_j$ 时,那么就称向量 $\boldsymbol{u}$ 支配向量 $\boldsymbol{v}$,记为 $\boldsymbol{u} \prec \boldsymbol{v}$。这种特性被称为 Pareto 支配。

定义 2　有且仅有在搜索空间 S 中不存在任何向量 $\boldsymbol{x} \in S$ 使得 $\boldsymbol{f}(\boldsymbol{x}) \prec \boldsymbol{f}(\boldsymbol{x}^*)$,那么就称决策向量 $\boldsymbol{x}^* \in S$ 为 Pareto 最优解,或者称为非支配解。

定义 3　在决策空间 S 内的 Pareto 最优解的集合被称为 Pareto 最优解集,记为 $\boldsymbol{P}^*$。Pareto 最优解集对应的目标函数值称为 Pareto 最优前沿,为 $F^* = \{\boldsymbol{f}(\boldsymbol{x}) : \boldsymbol{x} \in \boldsymbol{P}^*\}$。

1.2　基本粒子群算法

粒子群优化算法(particle swarm optimization, PSO)最初是为了模拟鸟群的飞行觅食行为而提出的。其初始化为一群随机粒子,通过迭代寻找最优解。在每一次迭代中,粒子通过跟踪 2 个极值来更新自己:①粒子本身所找到的最优解,称为个体最优值 p_{best};②整个种群目前找到的最优解,该极值称为全局最优值 g_{best}。在找到上述 2 个最优值时,粒子根据(2)式和(3)式来更新自己的速度和位置:

$$v_{id}^{k+1} = v_{id}^{k} + c_1 r_1 (p_{\mathrm{best}\ id}^{\ k} - x_{id}^{k}) + c_2 r_2 (g_{\mathrm{best}\ id}^{\ k} - x_{id}^{k}) \tag{2}$$

$$x_{id}^{k+1} = x_{id}^{k} + v_{id}^{k+1} \tag{3}$$

式中,c_1 和 c_2 分别为认知和社会加速系数,取值范围为[0,4],一般取为 $c_1 = c_2 = 2$;r_1 和 r_2 为 2 个在[0,1]内服从均匀分布的随机变量。

1.3　多目标粒子群(CDMOPSO)算法

Coello 在 2002 年首次提出了多目标粒子群(MOPSO)算法[11],其独特的搜索机理使其特别适合处理多目标优化问题。在此基础上,Raqual 于 2005 年提出了 CDMOPSO(crowding distance multi-objective PSO)算法[12],CDMOPSO 将拥挤距离算子(见图 1)[13]引入 PSO 算法中,用于选择引导粒子运动的最优解以及维护存储非支配解的外部档案,增加了整个种群的多样性,从而避免算法在早期就陷入局部最优。此算法的优点是结构简单,易于实现。其具体流程见图 2。

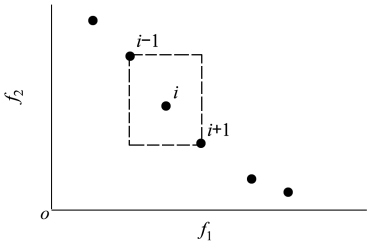

图 1　拥挤距离计算示意图

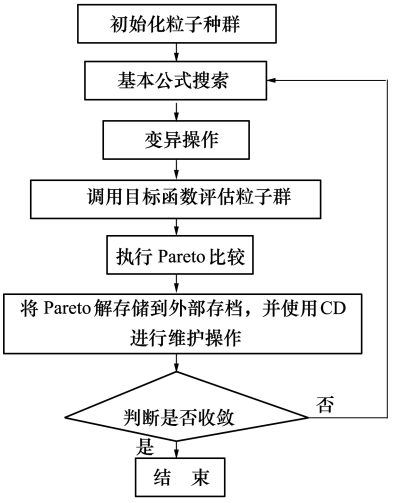

图 2　CDMOPSO 流程图

2 ASMOPSO 算法

2.1 α-stable 分布

α-stable 分布是一个四参数分布函数族[14-15]是一类广泛应用的随机信号模型,记为 $S(\alpha,\beta,\gamma,\sigma)$,其中,$\alpha \in (0,2]$ 是稳定性参数,$\beta \in [-1,1]$ 是偏斜参数,$\gamma \in (0,\infty)$ 是放大系数,$\sigma \in (-\infty,\infty)$ 是位移参数, 参数 α 和 β 决定了 α-stable 分布的形状。

α-stable 分布是一个非常丰富的分布族,它有 3 种特殊情况:

1) 当 $\alpha = 2$ 时是高斯分布,记为 $S\left(2,\beta,\frac{\gamma}{\sqrt{2}},\sigma\right)$,此时与偏斜参数 β 不相关;

2) 当 $\alpha = 1$,且 $\beta = 0$ 时为柯西分布,记为 $S(1,0,\gamma,\sigma)$;

3) 当 $\alpha = 0.5$ 且 $\beta = 1$ 时是 Levy 分布,记为 $S(0.5,1,\gamma,\sigma)$。

它的概率密度函数可以用特征函数的连续傅里叶变换来定义

$$f(x;\alpha,\beta,\gamma,\sigma) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \exp \cdot [it\sigma - |\gamma t|^{\alpha}(1 - i\beta \operatorname{sgn}(t))\Phi]e^{-itx}dt \quad (4)$$

式中, $\operatorname{sgn}(t)$ 是 t 的符号,Φ 表示为

$$\Phi = \begin{cases} \tan(\pi\alpha/2), \alpha \neq 1 \\ -(2/\pi)\ln|t|, \alpha = 1 \end{cases} \quad (5)$$

2.2 α-stable 变异

在 α-stable 分布中,稳定性系数 α 是一个非常重要的参数,它描述了分布的尾迹大小,决定了生成的随机数分布范围。图 3 给出了不同 α 值对应的 α-stable 分布。

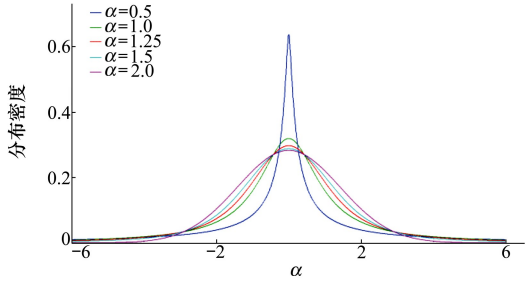

图 3 α 对分布密度的影响

本文的 PSO 算法中采用 α-stable 分布产生随机数,对 PSO 种群中的个体进行变异操作。变异过程中通过动态调整 α-stable 函数的稳定性系数 α 来实现变异范围和幅度的变化,变化范围为[1,2],其变化过程见图 4。在 PSO 程序循环的开始阶段,使用较小的 α 值,此时变异能力强,有助于增强算法的全局搜索能力,避免算法过早陷入局部最优;随着 α 值逐渐变大,全局变异能力减弱,但是局部搜索能力增强;最终 α 值增大为 2,此时执行的是传统的高斯变异,局部搜索能力最强,有利于提高解的精度。这种动态 α-stable 变异可以很好地平衡算法的开发和探索能力,使得改进后的多目标粒子群优化算法 ASMOPSO 算法兼顾精度和全局搜索能力,具有更好的适用性。α-stable 变异操作的伪代码见图 5。

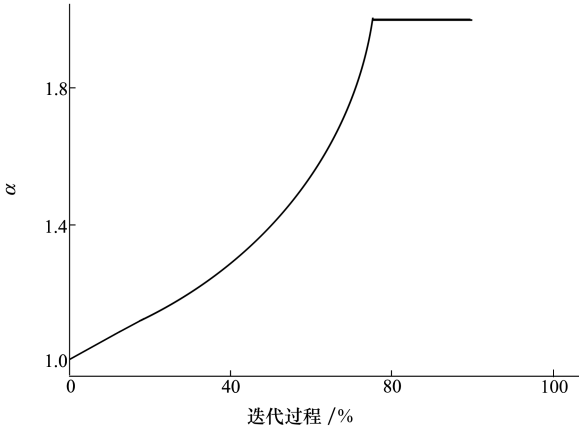

图 4 α 值的变化过程

```
%positon=particle position to be mutated
%dims=number of dimension
%currentgen=current iteration
%totgen=total number of iterations
%mutrate=mutation rate
%α=α value
function Mutation-Operator(position,dims,currentgen,totgen,mutrate)
  begin
    if rand<mutrate*(1-currentgen/totgen)^5/mutrate then
      begin
        for i=1 to n_dims
          position(i)=position(i)+(upbound(i)-lowbound(i))*s(α,0,
mutrate,0)
          if position(i)>upbound(i))then position(i)=upbound(i)
%set fly boundary of particle
          if position(i)<lowbound(i)) then position(i)=lowbound(i)
        end for
      end if
end function
```

图 5 α-stable 变异伪代码

2.3 本文提出的算法(ASMOPSO)

本文提出的 ASMOPSO 算法是在 CDMOPSO 算

法的基础上实现的,其详细实现步骤如下:

步骤 1　设置粒子群种群 POP 规模为 N,加速系数 c_1 和 c_2,惯性系数 w,外部档案 A 容量 N_{REP},变异概率 p_{mut},给定最大迭代步数 $I_{\max}$,设置外部档案 $A=\varphi$;

步骤 2　对于种群中的每个粒子($i=1,\cdots,N$),随机初始化位置 $\boldsymbol{X}_i$,初始速度设置为 $\boldsymbol{V}_i=0$,计算初始目标向量值 $\boldsymbol{f}(\boldsymbol{X}_i)$,初始化粒子的最优位置 $\boldsymbol{p}_i=\boldsymbol{X}_i$;

步骤 3　令 $t=0$;对种群中所有粒子进行 Pareto 比较,将非支配粒子储存在外部档案 A 中;

步骤 4　$t=t+1$;计算外部档案 A 中的非支配解的拥挤距离,将非支配解按照拥挤距离值大小降序排列;

步骤 5　对于 POP 中的每个粒子($i=1,\cdots,N$):从外部档案 A 中顶部一定比例(例如 10%)的非支配解中随机选择一个作为全局最优粒子 $\boldsymbol{p}_g$;用公式(2)和(3)更新粒子的速度和位置;如果 $\boldsymbol{X}_i$ 超出搜索空间边界,那么就设置粒子的位置为边界值,同时将速度乘以(-1),使粒子反向移动;使用 2.2 节的方法对粒子执行 α-stable 变异操作;计算粒子的目标向量值 $\boldsymbol{f}(\boldsymbol{X}_i)$;更新粒子个体最优位置;

步骤 6　对种群 POP 中所有粒子进行 Pareto 比较,并将新的非支配粒子与外部档案 A 中非支配粒子进行 Pareto 比较,将外部档案 A 中的支配粒子删除;如果外部档案 A 中的非支配解的数量等于 N_{REP},那么进行下列操作:①计算档案 A 中的每个非支配粒子的拥挤距离;②将外部档案中的个体按照拥挤距离值降序排列;③从 A 中最拥挤区域(拥挤距离值最小的 10%)中随机选择一个,然后将其用新的非支配解代替;

步骤 7　如果 $t=I_{\max}$,结束程序,否则转到步骤 4。

3　基准函数测试

3.1　评价指标

要对多目标优化算法的性能进行全面的量化分析,需要从它们的收敛性能、解集分布的多样性和宽广性三方面进行比较。本文使用反转世代距离(inverted generation distance,IGD)来综合评价算法的性能,IGD 既能评价多目标解集的收敛性,又能在一定程度上评价解集的分布特性。其定义如下

$$A_{\mathrm{IGD}}=\frac{\sum_{v\in \boldsymbol{P}^*} d(v,\boldsymbol{P})}{\boldsymbol{P}^*} \tag{6}$$

式中,$d(v,\boldsymbol{P})$ 表示近似解集 $\boldsymbol{P}$ 中的个体到真实前沿 $\boldsymbol{P}^*$ 中个体 v 之间的最小欧氏距离;该算法求得的最优解集用 $\boldsymbol{P}$ 表示。IGD 指标值越低,优化结果越理想,算法性能越优异。

3.2　参数设置

本文选择 ZDT1、ZDT2、ZDT3、ZDT4、Tanaka 以及 Srinivas 6 个测试函数来评估 ASMOPSO 算法的性能[16-18],并与 CDMOPSO 算法进行对比。

表 1 给出了 6 个基准函数的基本属性。6 个测试函数中,ZDT 系列函数属于无约束的函数,而 Tanaka 和 Srinivas 属于带有约束的多目标测试函数。ZDT 系列测试函数的设计变量维度高,优化难度大,特别是 ZDT3 函数的 Pareto 前沿是非连续的,ZDT4 函数是一个多峰值函数,拥有多个局部最优前沿。Tanaka 的 Pareto 前沿曲线被分为三段,其中中间段的水平和垂直部分由于解之间的差异比较小,是最难搜索的部分。Srinivas 是一个具有 2 个设计变量和二目标带约束的优化问题,但其具有的 2 个非线性约束项给优化问题还是带来了一定的困难。CDMOPSO 和 ASMOPSO 算法的初始种群大小均取为 100,最大循环次数为 200,惯性权值 w 从 0.9 线性递减为 0.4,加速系数 $c_1=c_2=2.0$,变异概率为0.4。

表 1　基准测试函数

函数名称	维数	变量范围	函数目标与约束	注解
ZDT1	30	[0,1]	$\min f_1(\boldsymbol{x})=x_1$ $\min f_2(\boldsymbol{x})=g\left(1-\sqrt{\left(\frac{f_1}{g}\right)}\right)$ $g(\boldsymbol{x})=1+9\sum_{i=2}^{n}\frac{x_i}{(n-1)}$	凸函数

续表 1

函数名称	维数	变量范围	函数目标与约束	注解
ZDT2	30	$[0,1]$	$\min f_1(\boldsymbol{x}) = x_1$ $\min f_2(\boldsymbol{x}) = g\left(1-\left(\frac{f_1}{g}\right)^2\right)$ $g(\boldsymbol{x}) = 1+9\sum_{i=2}^{n}\frac{x_i}{(n-1)}$	凹函数
ZDT3	30	$[0,1]$	$\min f_1(\boldsymbol{x}) = x_1$ $\min f_2(\boldsymbol{x}) = g\left(1-\sqrt{\frac{f_1}{g}}-\left(\frac{f_1}{g}\right)\sin(10\pi f_1)\right)$ $g(\boldsymbol{x}) = 1+9\sum_{i=2}^{n}\frac{x_i}{(n-1)}$	非连续凸函数
ZDT4	10	$x_1 \in [0,1]$ $x_i \in [-5,5]$ $i = 2,\cdots,n$	$\min f_1(x) = x_1$ $\min f_2(x) = g\left(1-\sqrt{\frac{f_1}{g}}\right)$ $g(\boldsymbol{x}) = 1+10(n-1)+\sum_{i=2}^{n}(x_i^2-10\cos(4\pi x_i))$	多模态凸函数
Srinivas	2	$[-20,20]$	$\min f_1(x) = 2+(x_1-2)^2+(x_2-1)^2$ $\min f_2(x) = 9x_1-(x_2-1)^2$ s.t. $e_1(x) = x_1^2+x_2^2 \leqslant 225$ $e_2(x) = x_1-3x_2+10 \leqslant 0$	凸函数
Tanaka	2	$(0,\pi]$	$\min\ f_1(x,y) = x$ $\min\ f_2(x,y) = y$ s.t. $-x^2-y^2+1+0.1\cos\left(16\arctan\frac{x}{y}\right) \leqslant 0$ $\left(x-\frac{1}{2}\right)^2+\left(y-\frac{1}{2}\right)^2-\frac{1}{2} \leqslant 0$	非凸不连续函数

3.3 函数测试结果分析

对每个基准函数进行 30 次独立运行之后的 IGD 值统计结果列于表 2,表中 AVE 表示平均值,STD 表示标准差。从表中数据可以看出,本文提出的 ASMOPSO 在 4 个 ZDT 函数上的性能都优于 CDMOPSO,尤其在优化难度较大的 ZDT3 和 ZDT4 函数上的优势明显;对于 2 个带约束的函数,CDMOPSO 的 IGD 均值都略小于 ASMOPSO,但是差距都非常小。

表 2 CDMOPSO 算法与 ASMOPSO 算法的 IGD 指数比较

	算法	ZDT1	ZDT2	ZDT3	ZDT4	Tanaka	Srinivas
CDMOPSO	AVE	4.192×10^{-3}	4.170×10^{-3}	1.605×10^{-1}	4.675	4.868×10^{-3}	8.577×10^{-1}
	STD	8.967×10^{-5}	8.877×10^{-5}	7.706×10^{-2}	2.959	2.783×10^{-4}	1.424×10^{-2}
ASMOPSO	AVE	3.956×10^{-3}	3.942×10^{-3}	1.166×10^{-2}	7.381×10^{-3}	5.512×10^{-3}	8.786×10^{-1}
	STD	3.631×10^{-5}	4.299×10^{-5}	1.170×10^{-2}	7.317×10^{-3}	4.033×10^{-4}	1.615×10^{-2}

图 6 和图 7 给出了 2 种算法任意一次优化搜索到的近似 Pareto 解与真实 Pareto 前沿的比较。可以看出改进后的 ASMOPSO 算法获得非支配解与真实 Pareto 前沿的偏差很小,而且都均匀地分布在真实前沿上,特别是在求解 ZD4 测试函数时表现出了非常优异的性能。图 9 和图 10 表明 CDMOPSO 在处理 ZDT4 这种多模态问题时寻优能力欠佳,同时 CDMOPSO 生成的 ZDT3 前沿也没有均匀地分布在真实前沿上,且 Pareto 解的数量明显偏少。

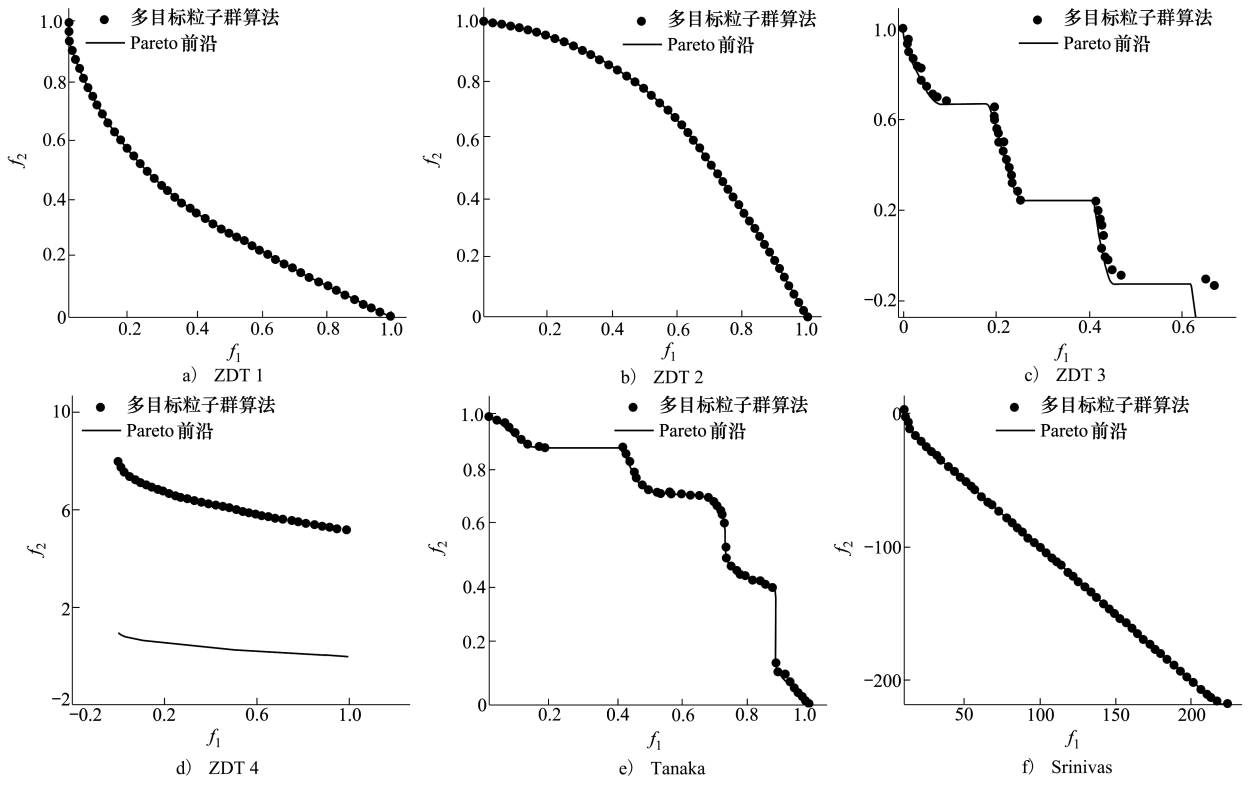

图 6　CDMOPSO 测试函数结果

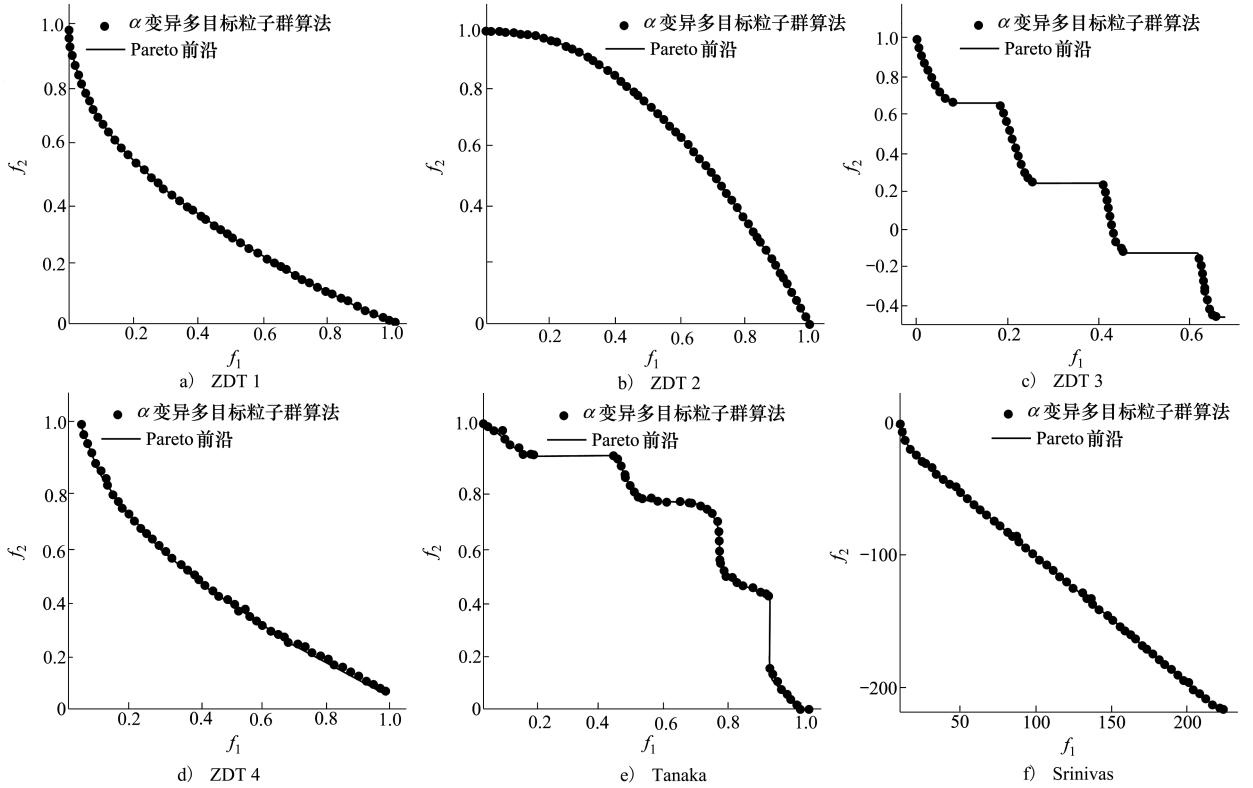

图 7　ASMOPSO 测试函数结果

4 工程应用

为了进一步验证本文提出的多目标粒子群算法的优化性能,将其应用于 RAE2822 翼型跨音速优化设计。

4.1 优化模型的建立

以 RAE2822 翼型在 $Ma=0.73$, $\alpha=2.79°$, $Re=6.5\times10^6$ 时最小化阻力系数和俯仰力矩系数绝对值不减小为目标进行外形优化设计[19]。约束条件为升力系数大于原始翼型的升力系数值,同时保持翼型的最大厚度不减小。由此可得优化的数学模型为

$$\begin{cases}\min f_1(\boldsymbol{X})=C_d\\ \quad f_2(\boldsymbol{X})=|C_m|\leqslant|C_m^*|\\ \text{s.t. } g_1(\boldsymbol{X})=C_l\geqslant C_l^*\\ \quad g_2(\boldsymbol{X})=c_{\max}\geqslant c^*\end{cases}\tag{7}$$

式中,$\boldsymbol{X}$ 是设计变量向量,C_d 是阻力系数,C_m 是力矩系数,C_l 是升力系数,$c_{\max}$ 是翼型的最大厚度,上标 * 表示初始翼型的计算值。

本文的翼型参数化使用型函数为 6 阶 Bernstein 多项式的 CST 方法。翼型上下表面各取 6 个设计参数,共 12 个优化设计变量。图 8 ~ 11 显示了 RAE2822 初始翼型的 CST 参数化结果。从图中可以看出,CST 参数化后的上翼面外形与原翼型几乎完全重合,而下翼面在 $x/c=0.1\sim0.5$ 段误差较大。从数值上看上翼面的最大拟合误差小于 0.000 2,下翼面的最大拟合误差小于 0.000 5,而在工程中对于翼型最大拟合误差小于 0.001(弦长为 1 时),则认为该参数化方法能精确拟合当前翼型。从图 12 可知经过 CST 参数化后的拟合翼型表面的 C_p 分布基本与原翼型重合,下表面有较小的差异,主要集中在 $x/c=0.2\sim0.5$ 这一段,上表面的 C_p 分布的差异较大,在前缘附近未能很好的捕捉负压鼓包区,在翼型中部,激波出现的比原始翼型稍早,但是差异不是很明显;而表 3 则直观说明了 2 种翼型的气动力特性差别非常小,升力系数和阻力系数的相对误差都小于 1%,力矩系数误差略大,为 1.6%。参数化结果表明本文选用的基于 6 阶 Bernstein 多项式的 CST 方法可以较为精确地描述 RAE2822 翼型的设计空间。

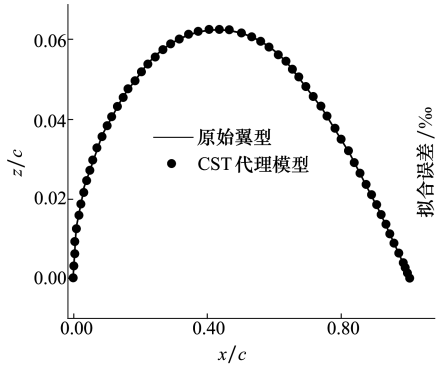

图 8 上翼面外形拟合结果

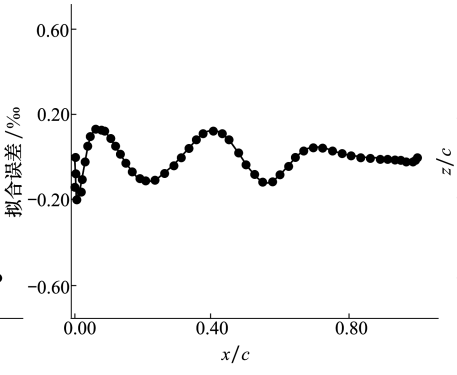

图 9 上翼面外形拟合误差

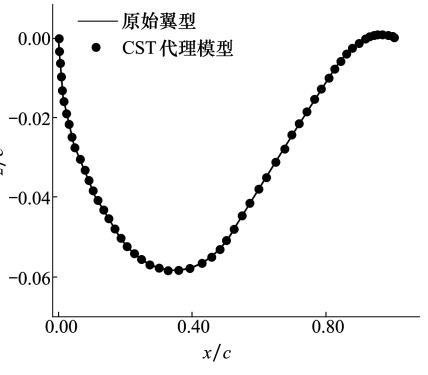

图 10 下翼面外形拟合结果

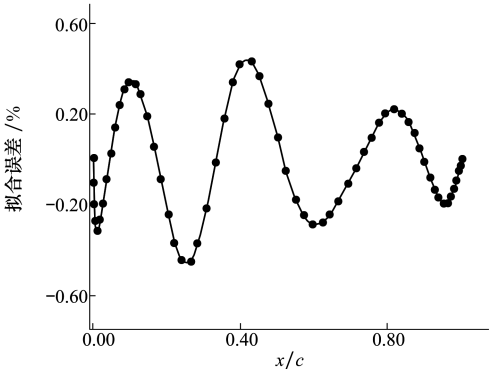

图 11 下翼面外形拟合误差

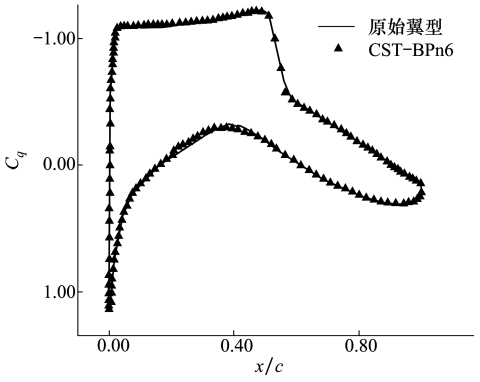

图 12 RAE2822 翼型 CST 参数化后压力系数比较

表 3　RAE2822 翼型参数化后力学特性结果比较

参数值	原始翼形	CST_BPn6	误差/%
C_l	0.758 0	0.754 6	−0.45
C_d	0.017 7	0.017 6	−0.56
C_m	−0.087 3	−0.085 9	−1.60

4.2　优化结果及分析

采用本文改进的 ASMOPSO 和 CDMOPSO 2 种方法对 RAE2822 翼型进行跨音速多目标优化设计并对比两者的优化效果。2 个优化算法的种群规模均为 100,两者的外部档案大小为 100;惯性系数为 0.4~0.9。认知加速系数和社会加速系数 $C_1 = C_2 = 2.0$。ASMOPSO 优化算法中的系数 α 变化范围为 1.0~2.0。2 种算法的迭代步数均为 100 步。

由图 13 可知,多次优化后,ASMOPSO 优化算法得到的非支配解集分布均匀,而且比 CDMOPSO 的优化结果更接近真实 Pareto 前沿。所以后续的对比中,仅需要与原始翼型对比即可。由于多目标解集更能客观的反应优化问题的真实内涵,选取点更能符合不同设计人员的需求,那么在均衡操纵性和减阻效果下,如图 13 所示,取 Pareto 前缘的中间一点。在此点下的气动数据如表 4 所示。图 14 与图 15 分别为原始翼型与优化后翼型的几何外形比较与 C_p 分布比较。可以看出优化后翼型,最大厚度位置向后移动,前缘压力峰值和上翼面中部的局部压力变化有明显的减缓。图 16 与图 17 的压力云图可以看出优化翼型激波明显减小。因此在此取点下的优化结果具有良好的阻力特性和力矩特性。

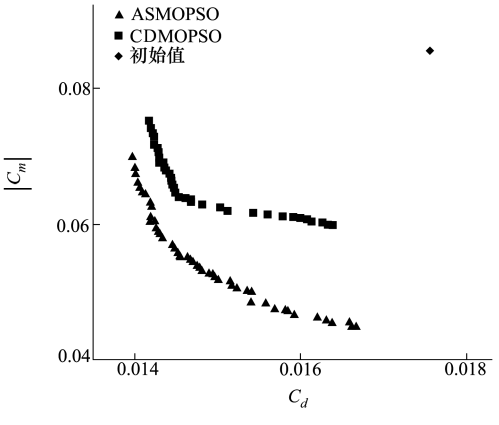

图 13　RAE2822 翼型优化后的 Pareto 前缘

表 4　优化前后的气动数据比较

参数值	初始值	ASMOPSO
C_l	0.754 5	0.754 9
C_d	0.017 6	0.014 6
$\lvert C_m \rvert$	0.085 8	0.055 1
$C_{\max}$	0.120 8	0.122 5

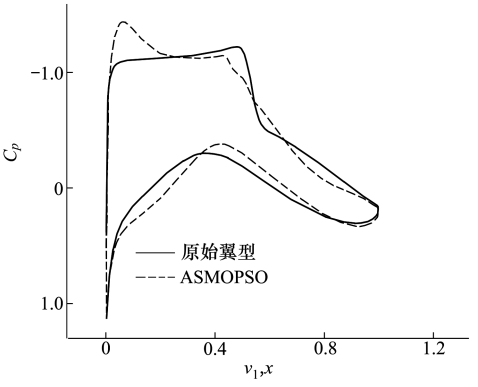

图 14　翼型表面压力分布对比

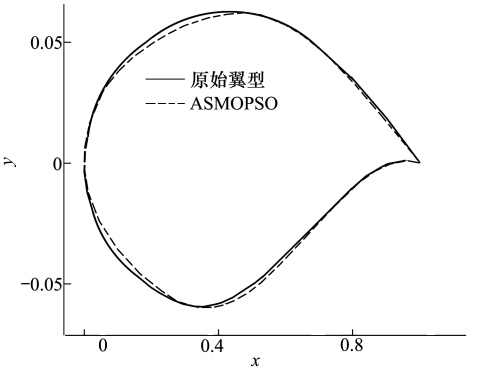

图 15　优化翼型外形对比

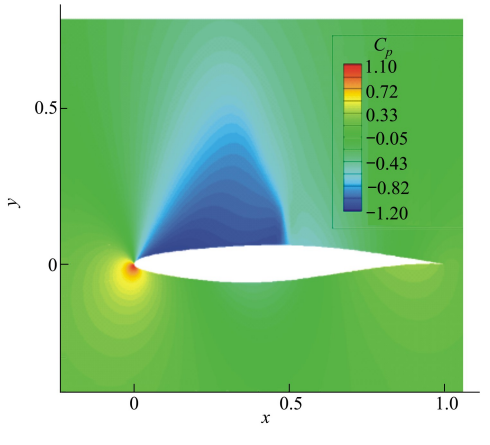

图 16　原始翼型压力云图

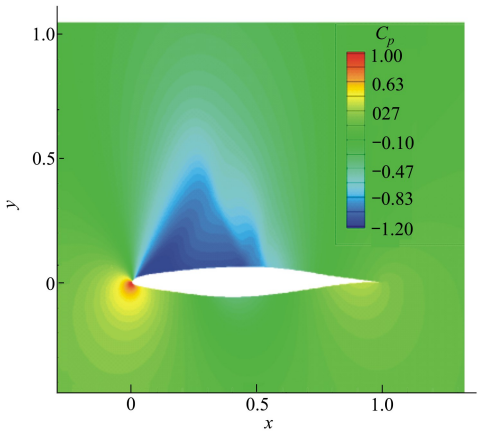

图 17 ASMOPSO 优化后翼型压力云图

5 结 论

本文针对 CDMOPSO 算法容易陷入局部最优的缺点,结合 α-stable 分布理论,建立了改进的 ASMOPSO 多目标优化算法。使用多个基准函数对比测试了改进前后的算法性能,并将改进算法应用到了 RAE2822 翼型跨声速多目标优化设计中,得到的结论有:

1) 引入 α-stable 分布理论在 PSO 算法中对种群进行变异操作,根据 α-stable 分布随稳定系数 α 来的特点,通过改变 α 值来动态调整优化过程中的变异强度和范围,使得改进后的 ASMOPSO 算法较原始算法具有更好的局部和全局搜索兼顾能力。

2) 无约束的 ZDT 系列函数和带约束的 Tanaka 和 Srinivas 的测试结果显示,改进后的算法能够有效引导非支配解逼近真实 Pareto 前沿,尤其是对多峰值 ZDT4 函数的求解,表明改进后的算法在处理多模态问题时有很强的寻优能力;

3) RAE2822 翼型的跨声速减阻和俯仰力矩系数绝对值不减小的综合优化设计中,新的 ASMOPSO 算法得到的结果明显优于 CDMOPSO。

参考文献:

[1] QU X, ZHANG R, LIU B, et al. An Improved TLBO Based Memetic Algorithm for Aerodynamic Shape Optimization[J]. Engineering Applications of Artificial Intelligence, 2017, 57: 1-15

[2] LEIFSSON L, KOZIEL S, TESFAHUNEGN Y A. Multiobjective Aerodynamic Optimization by Variable-Fidelity Models and Response Surface Surrogates[J]. AIAA Journal, 2016, 54: 531-541

[3] EBRAHIMI M, JAHANGIRIAN A. Accelerating Global Optimization of Aerodynamic Shapes Using a New Surrogate-Assisted Parallel Genetic Algorithm[J]. Engineering Optimization, 2017, 49(12): 1-16

[4] CAO L, ZHANG D. Aerodynamic Configuration Optimization for Hypersonic Gliding Vehicle Based on Improved Hybrid Multi-Objective PSO Algorithm[C]// IEEE International Conference on Signal Processing, Communications and Computing, 2015: 1-5

[5] SONG L, LUO C, LI J, et al. Aerodynamic Optimization of Axial Turbomachinery Blades Using Parallel Adaptive Range Differential Evolution and Reynolds-Averaged Navier-Stokes Solutions[J]. International Journal for Numerical Methods in Biomedical Engineering, 2011, 27(2): 283-303

[6] WEISHUANG L U, TIAN Y, LIU P. Aerodynamic Optimization and Mechanism Design of Flexible Variable Camber Trailing-Edge Flap[J]. Chinese Journal of Aeronautics, 2017, 30(3): 988-1003

[7] KOO D, ZINGG D W. Investigation into Aerodynamic Shape Optimization of Planar and Nonplanar Wings[J]. AIAA Journal, 2017, 56(1): 1-14

[8] 李丁,夏露. 一种混合粒子群优化算法在翼型设计中的应用[J]. 航空计算技术, 2010, 40(6):66-71
LI Ding, XIA Lu. Application of a Hybrid Particle Swarm Optimization to Airfoil Design[J]. Aeronautical Computing Technique, 2010, 40(6): 66-71 (in Chinese)

[9] 陈进, 郭小锋, 孙振业, 等. 基于改进多目标粒子群算法的风力机大厚度翼型优化设计[J]. 东北大学学报, 2016, 37(2):232-236
CHEN Jin, GUO Xiaofeng, SUN Zhenye, et al. Optimization of Wind Turbine Thick Airfoils Using Improved Multi-Objective Particle Swarm Algorithm[J]. Journal of Northeastern University, 2016, 37(2):232-236 (in Chinese)

[10] 李鑫, 屈转利, 李耿,等. 高效低噪的二维翼型优化设计[J]. 振动与冲击, 2017, 36(4): 66-72
LI Xin, QU Zhuanli, LI Geng, et al. A Numerical Optimization for High Efficiency and Low Noise Airfoils[J]. Journal of Vibra-

tion and Shock, 2017, 36(4): 66-72 (in Chinese)
[11] WERON A, WERON R. Computer Simulation of Levy-α Stable Variables and Processes[M]. Poland, Springer, Berlin Heidelberg, 1995: 379-392
[12] KOGON S M, MANOLAKIS D G. Signal Modeling with Self-Similar α Stable Processes: The Fractional Levy Stable Motion Model[J]. IEEE Trans on Signal Processing, 1996, 44(4): 1006-1010
[13] WEI J L, JAMBEK A B, NEOH S C. Kursawe and ZDT Functions Optimization Using Hybrid Micro Genetic Algorithm(HMGA)[J]. Soft Computing-a Fusion of Foundations, Methodologies and Applications, 2015, 19(12):3571-3580
[14] LI X. Better Spread and Convergence: Particle Swarm Multiobjective Optimization Using the Maximin Fitness Function[C]// Genetic and Evolutionary Computation Conference, 2004: 117-128
[15] PENG H, LI R, CAO L L, et al. Multiple Swarms Multi-Objective Particle Swarm Optimization Based on Decomposition[J]. Procedia Engineering, 2011, 15(2): 3371-3375
[16] PENG X, LIU D, SHAN J, et al. Airfoil Aerodynamic Optimization Based on an Improved Genetic Algorithm[C]// International Conference on Intelligent Systems Design & Engineering Applications, 2014: 133-137
[17] Uoigne Alan Le, Qin Ning. Variable-Fidelity Aerodynamic Optimization for Turbulent Flows Using a Discrete Adjoint Formulation[J]. AIAA Journal, 2004, 42(42):1281-1292
[18] LAURENCEAU J, MEAUX M, MONTAGNAC M, et al. Comparison of Gradient-Based and Gradient-Enhanced Response-Surface-Based Optimizers[J]. AIAA Journal, 2010, 48(5):981-994
[19] DEB K, PRATAP A, AGARWAL S, et al. A Fast and Elitist Multi-Objective Genetic Algorithm: NSGA-II [J]. IEEE Trans on Evolutionary Computation, 2002, 6(2): 182-197

Research and Application of Multi-Objective Particle Swarm Optimization Algorithm Based on α-Stable Distribution

FAN Huayu, ZHAN Hao, CHENG Shixin, MI Baigang
(School of Aeronautics, Northwestern Polytechnical University, Xi'an 710072, China)

Abstract: The multi-objective particle swarm optimization algorithm (MOPSO) has been applied and modified for optimal designs in various engineering fields. However, the intimal algorithm is still has the problems of low accuracy and poor diversity of solutions as it is easy to fall into local optimum in the later evolution stage. A new dynamic mutation operator has been established based on the α-stable distribution theory and incorporated with multi-objective particle swarm optimization algorithm(ASMOPSO). By using random Numbers which generated by the α-stable distribution, the population of PSO algorithm was mutated. And this mutate operation increases the diversity of the population. Because the stability coefficient in the ASMOPSO algorithm can change the range and amplitude of the mutation. This operation makes the new algorithm has the ability to balance the calculation accuracy and global optimization. Several benchmark functions test show that the ASMOPSO algorithm has fast global optimization ability. The proposed algorithm is applied to the multi-objective aerodynamic optimization design of RAE2822 transonic airfoil. The comparison results also show that ASMOPSO algorithm is more excellent than the basic MOPSO algorithm.

Keywords: multi-objective particle swarm optimization; α-stable distribution; dynamic mutation; airfoil design; aerodynamic optimization

© 2019 Journal of Northwestern Polytechnical University.

This is an Open Access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.