

文章编号: 1674-8190(2023)03-050-11

# 超参数自适应的MOEA/D-DE算法在翼型气动隐身优化中的应用

王培君, 夏露, 栾伟达, 陈会强  
(西北工业大学 航空学院, 西安 710072)

**摘要:** MOEA/D-DE算法易于实现, 被广泛应用于处理多目标优化问题, 但其超参数对算法性能影响较大。基于MOEA/D-DE算法框架, 利用Sobol全局灵敏度分析方法对差分进化算子中的交叉控制参数进行改进, 使用莱维飞行机制控制比例因子, 使算法中的超参数拥有自适应能力, 得到超参数自适应的MOEA/D-DE算法——MOEA/D-DEAH算法; 对MOEA/D-DEAH算法、不同超参数设置的MOEA/D-DE算法和NSGA II算法进行函数测试和翼型气动隐身优化算例对比。结果表明: MOEA/D-DEAH算法性能良好, 具有较强的鲁棒性, 气动隐身优化效果也比其他算法更好。

**关键词:** 多目标优化算法; 基于分解的多目标优化算法(MOEA/D); 超参数; 灵敏度分析; 气动隐身优化; 差分进化算子

中图分类号: V218; TP18

文献标识码: A

DOI: 10.16615/j.cnki.1674-8190.2023.03.05

## Application of hyper-parameters adaptive MOEA/D-DE algorithm in aerodynamic stealth optimization of airfoil

WANG Peijun, XIA Lu, LUAN Weida, CHEN Huiqiang  
(School of Aeronautics, Northwestern Polytechnical University, Xi'an 710072, China)

**Abstract:** MOEA/D-DE algorithm is easy to implement and widely used to deal with multi-objective optimization problems, while its hyper-parameters have a great impact on the performance of the algorithm. In this paper, based on the MOEA/D-DE algorithm framework, the Sobol global sensitivity analysis method is used to improve the cross control parameter in the differential evolution operator, and the Levy flight strategy is used to control the scale factor. Then the hyper-parameters in the MOEA/D-DE algorithm obtain adaptive ability. And MOEA/D-DE algorithm with adaptive hyper-parameters (MOEA/D-DEAH) is proposed. The MOEA/D-DEAH MOEA/D-DE with different hyper-parameters settings and NSGA II are tested and compared with function tests and aerodynamic stealth optimization of airfoil. The function test results show that the new algorithm has good performance, and the algorithm can obtain strong robustness by adapting the super parameters. The optimization results of airfoil aerodynamic stealth show that MOEA/D-DEAH algorithm has better optimization results than other algorithms.

**Key words:** multi-objective optimization algorithm; multiobjective evolutionary algorithm based on decomposition (MOEA/D); hyper-parameters; sensitivity analysis; aerodynamic stealth optimization; differential evolution operator

收稿日期: 2022-05-26; 修回日期: 2022-10-09

基金项目: 翼型叶栅空气动力学重点实验室基金(6142201200106)

通信作者: 夏露, xialu@nwpu.edu.cn

引用格式: 王培君, 夏露, 栾伟达, 等. 超参数自适应的MOEA/D-DE算法在翼型气动隐身优化中的应用[J]. 航空工程进展, 2023, 14(3): 50-60.

WANG Peijun, XIA Lu, LUAN Weida, et al. Application of hyper-parameters adaptive MOEA/D-DE algorithm in aerodynamic stealth optimization of airfoil[J]. Advances in Aeronautical Science and Engineering, 2023, 14(3): 50-60. (in Chinese)

## 0 引言

飞行器在进行外形设计时需要同时考虑其气动性能和隐身性能等多个设计要求,这就涉及到多目标优化问题(Multiobjective Optimization Problem,简称MOP)的求解。对于多目标优化问题,一般需要寻找一组被称为Pareto最优解集的解。较简单的处理方法是通过对多个目标处理为一个目标,再使用单目标优化算法进行寻优。夏露等<sup>[1]</sup>使用这种方法进行了飞翼布局的翼型优化设计,但该方法每次运行只能得到一个Pareto解,无法一次获得不同权重的解,具有一定局限性。多目标优化算法可以在一次运行中产生一组Pareto解,张彬乾等<sup>[2]</sup>使用多目标算法对飞翼布局各站位翼型开展了气动隐身多目标优化。多目标优化算法近年来得到了蓬勃发展,常用的算法有:基于拥挤距离的非支配排序遗传算法(Nondominated Sorting Genetic Algorithm II,简称NSGA II)<sup>[3]</sup>、基于分解的多目标优化算法(Multiobjective Evolutionary Algorithm Based on Decomposition,简称MOEA/D)<sup>[4]</sup>、基于指标的多目标优化算法(Indicator-based Evolutionary Algorithm,简称IBEA)<sup>[5]</sup>等。

在多目标优化算法中,以MOEA/D为代表的算法由于其基于分解的机理可以获得具有较好分布性、均匀性、收敛性的解,性能优越。目前,对MOEA/D算法及其他基于分解的算法的改进主要集中在权重向量的选择和标化方法上。Q. Liu等<sup>[6]</sup>对增长神经气体网络(Growing Neural Gas,简称GNG)进行改进得到了iGNG(improved Growing Neural Gas),并用其来适应不规则Pareto前缘,得到了RVEA-iGNG(iGNG-based Reference Vector Adaptation for Evolutionary Optimization)算法,该算法可以有效捕捉真实Pareto前缘,性能较好;S. Jiang等<sup>[7]</sup>提出了一种新的标化方法MSF(Multiplicative Scalarizing Function),并将其应用在MOEA/D上得到了eMOEA/D(efficient MOEA/D),函数测试表明,新算法的性能优异,而算法中的算子及其超参数对算法性能同样重要。对上述算法现有的研究仍较少,Li H等<sup>[8]</sup>通过将MOEA/D算法中的遗传(Genetic Algorithm,简称GA)算子替换成差分进化(Differential Evolution,简称DE)变异算子得到了MOEA/D-DE算法,该

算法能够处理复杂的Pareto解集,性能有了进一步提高,被广泛应用。但是MOEA/D-DE算法受算子控制参数影响较大,尤其对两个主要控制参数敏感:交叉控制参数 $C_r$ 和比例因子 $F$ 。交叉控制参数 $C_r$ 的取值与Pareto解集的复杂度呈正相关<sup>[9-10]</sup>,比例因子 $F$ 则控制着算法的局部搜索能力与全局搜索能力。这些参数的选择依赖于使用者的经验,且在求解问题之前使用者无法预先得知问题的相关特征,故无法对参数进行针对性调整,这给处理工程优化问题带来了一定困难。朱新奇<sup>[11]</sup>提出了局部坐标交叉算子改善的MOEA/D-DE算法,减小了对 $C_r$ 的依赖,但仍有一定的依赖性。

为了解决MOEA/D-DE算法超参数选择困难的问题,本文基于MOEA/D-DE算法框架,通过Sobol灵敏度分析<sup>[12]</sup>来提取问题中更多的信息,对DE变异算子超参数进行自适应改进。首先通过Sobol灵敏度分析获得多目标问题中每个自变量对于不同问题的灵敏度;其次根据“自变量灵敏度越高对问题越重要、需要进行的变异次数越多”的原则对交叉控制参数 $C_r$ 进行修改,使 $C_r$ 值根据自变量不同与子问题目标权重不同进行自适应;然后利用莱维(Levy)飞行机制对比例因子 $F$ 进行改进,使算法可以兼顾全局与局部搜索;最后得到自适应超参数的MOEA/D-DE算法——MOEA/D-DEAH(Multiobjective Evolutionary Algorithm Based on Decomposition Algorithm with Adaptive Hyper-parameters),通过函数测试验证算法效果,并在翼型气动隐身优化中,将新算法与其他算法进行比对,进一步验证算法的优势。

## 1 MOEA/D-DE 算法

### 1.1 MOEA/D 算法

基于分解的多目标优化算法(MOEA/D)用分解方法将多目标问题分解为若干个子优化问题<sup>[4]</sup>,再用EA(进化算法)来优化这些子问题,经过一次优化就可得到一组Pareto近似解。根据各子问题间加权向量的欧式距离来定义所有子问题之间的邻域关系,而相邻的两个子问题间具有相近的最优解,因此子问题可以利用其相邻子问题的当前信息来进行优化。MOEA/D算法的基本步骤如下:

1) 首先生成  $N$  个均匀分布的权重向量, 计算任意两个加权向量之间的欧氏距离, 并构造一个含有  $N$  个个体的种群;

2) 在对第  $i$  个子问题进行优化时, 根据第 1) 步中获得的欧氏距离确定该子问题的  $T$  个相邻子问题, 同时将相邻子问题的个体作为子种群;

3) 从子种群中选择个体使用遗传算子生成新个体;

4) 比较新个体函数值与其子问题的目标函数值, 并对邻近子问题的个体进行更新;

5) 更新外部种群;

6) 满足停止条件后, 结束并输出外部种群, 否则返回步骤 2)。

本文使用的分解方法为加权切比雪夫方法 (Weighted Tchebycheff Approach), 该分解方法的形式如下:

$$\min g^t(x|\mathbf{w}) = \max_{i=1, \dots, m} [\omega_i |f_i(x) - z_i^*|] \quad (x \in \Omega) \quad (1)$$

式中:  $\mathbf{w}$  为权重系数向量,  $\mathbf{w} = [\omega_1, \dots, \omega_m]^T$ , 满足  $\omega_i \geq 0$ , 且  $\sum_{i=1}^m \omega_i = 1$ ;  $\mathbf{z}^*$  为理想参考点 (Ideal Reference Point),  $\mathbf{z}^* = [z_1^*, \dots, z_m^*]^T$ , 即  $z_i^* = \min \{f_i(x) | x \in \Omega\}$ ,  $i = 1, \dots, m$ 。

因为 MOEA/D 算法是将多目标问题分解为若干个子优化问题, 所以不需要对种群中的解进行非支配排序, 也无需进行较为耗时的超体积计算。在保证运行速度的同时可以获得较好的种群多样性。

$$\sigma_j = \begin{cases} \left[ 2\mu + (1 - 2\mu)(1 - \delta_1)^{\eta_m + 1} \right]^{\frac{1}{\eta_m + 1}} - 1 & (\mu \leq 0.5) \\ 1 - \left[ 2(1 - \mu) + 2(\mu - 0.5)(1 - \delta_2)^{\eta_m + 1} \right]^{\frac{1}{\eta_m + 1}} & (\mu > 0.5) \end{cases} \quad (5)$$

式中:  $b_j$  与  $a_j$  分别为  $j$  维的设计空间上下限;  $p_m$  为突变率,  $p_m = 1/D$ ;  $\mu$  为  $0 \sim 1$  之间均匀分布的随机数;  $\eta_m$  为变异参数, 取值为 20。

MOEA/D-DE 算法相比 MOEA/D 算法在提高种群多样性上做出了以下两点改进:

① 在生成新个体时, MOEA/D-DE 算法允许 DE 算子以一定的概率从整个种群中选择父代个体, 由此可以扩大搜索范围, 提高种群的多样性;

② 在使用新个体对邻近子问题的个体进行更新时, MOEA/D-DE 限制更新个数, 从而提高了种

## 1.2 MOEA/D-DE 算法

MOEA/D 算法处理具有复杂 Pareto 解集的多目标优化问题时能力不足<sup>[8]</sup>, 使用 DE 算子后再采用突变算子生成新个体的 MOEA/D-DE 算法, 以提高算法处理复杂优化问题的能力。

DE 算子包含两个基本操作——突变和交叉。

突变操作公式为

$$\mathbf{v}_i = \mathbf{x}_{r_1} + F \cdot (\mathbf{x}_{r_2} - \mathbf{x}_{r_3}) \quad (i \in 1, 2, \dots, N_P) \quad (2)$$

式中:  $N_P$  为种群大小;  $r_1, r_2, r_3$  为在  $[1, N_P]$  中随机抽取的不同的整数;  $F$  为比例因子;  $\mathbf{v}$  为突变向量;  $\mathbf{x}$  为个体向量。

交叉操作结合突变向量  $\mathbf{v}_i$  和目标向量  $\mathbf{x}_i$ , 生成试验向量  $\mathbf{u}_i$ , 如式 (3) 所示。

$$\bar{u}_{i,j} = \begin{cases} v_{i,j} & (\text{rand}_j(0, 1) \leq C_R \text{ 或 } j = j_{\text{rand}}) \\ x_{i,j} & (\text{其他}) \end{cases} \quad (3)$$

式中:  $j_{\text{rand}}$  为介于 1 和设计变量数  $D$  之间的随机整数, 使得试验向量与目标向量至少相差一维;  $\text{rand}_j(0, 1)$  为  $0 \sim 1$  之间均匀分布的随机数;  $C_R$  为交叉控制参数。

MOEA/D-DE 算法生成新个体时不仅使用了 DE 算子, 还在 DE 算子之后使用了多项式变异算子。多项式变异算子为

$$u_{i,j} = \begin{cases} \bar{u}_{i,j} + \sigma_j \times (b_j - a_j) & (\text{rand}_j(0, 1) \leq p_m) \\ \bar{u}_{i,j} & (\text{其他}) \end{cases} \quad (4)$$

群的多样性。

但是 MOEA/D-DE 算法与其他算法一样, 算法性能受超参数的影响。为了消除这一影响, 需要从问题中获得更多信息来指导超参数的自动调节。本文使用 Sobol 灵敏度分析方法从优化问题中获取更多信息。

## 2 Sobol 灵敏度分析方法

Sobol 方法是一种基于 Monte Carlo 积分灵敏度分析方法<sup>[12]</sup>, 通过方差分解中的方差来表示灵

敏度。

设目标函数的输入输出可被描述为

$$Y=f(\mathbf{X}) \quad (6)$$

式中: $\mathbf{X}$ 为 $n$ 维输入变量, $\mathbf{X}=[X_1, X_2, \dots, X_n]$ ;  $Y$ 为一维输出变量。

设模型 $Y=f(\mathbf{X})$ 可以分解成如下形式:

$$f(\mathbf{X})=f_0+\sum_{i=1}^n f_i(X_i)+\sum_{i \neq j} f_{ij}(X_i, X_j)+\dots+f_{12\dots n}(X_1, X_2, \dots, X_n) \quad (7)$$

其中,

$$\begin{cases} f_0=E(Y) \\ f_i=E(Y/X_i)-f_0 \\ f_{ij}=E(Y/X_i, X_j)-f_i-f_j-f_0 \\ \dots \end{cases} \quad (8)$$

由式(7)~式(8)可知,模型函数可以分解为维数递增的 $2^n$ 个子函数项。若输入变量之间是相互独立的,则式(7)右端各项互相正交,因此对式(7)左右两端求方差可得:

$$V(Y)=\sum_{i=1}^n V_i+\sum_{i \neq j} V_{ij}+\sum V_{12\dots n} \quad (9)$$

其中,

$$\begin{cases} V_i=V[E(Y/X_i)] \\ V_{ij}=V[E(Y/X_i, X_j)]-V_i-V_j \\ V_{ijk}=V[E(Y/X_i, X_j, X_k)]-V_i-V_j-V_k \\ \dots \end{cases} \quad (10)$$

令 $S_i=V_i/V(Y)$ ,则:

$$S_i=\frac{V[E(Y/X_i)]}{V(Y)} \quad (11)$$

同理:

$$S_{ij}=V_{ij}/V(Y)=\frac{V[E(Y/X_i, X_j)]-V_i-V_j}{V(Y)} \quad (12)$$

以此类推,并对式(7)两端同时除以 $V(Y)$ ,可得:

$$\sum_{i=1}^n S_i+\sum_{i \neq j} S_{ij}+\sum S_{12\dots n}=1 \quad (13)$$

式中: $S_i$ 为单个变量 $X_i$ 对输出方差的贡献效果,也被称为主效应,同时也是Sobol方法中的一阶灵敏度指标; $S_{ij}$ 为变量 $X_i$ 和 $X_j$ 二者对方差的共同贡献再减去二者的一阶灵敏度,即变量 $X_i$ 和 $X_j$ 的交叉影响效果,在Sobol方法中也被称为二阶灵敏度指

标。同理可得更高阶数的灵敏度指标。

Sobol方法同时定义了总灵敏度指标 $S_{Ti}, S_{Ti}$ 代表的是 $X_i$ 的单独贡献与其他变量的交叉贡献的总和,其定义如式(14)所示。

$$S_{Ti}=S_i+\sum_{j=1, j \neq i}^n S_{ij}+\dots+S_{12\dots n}=1-S_{-i} \quad (14)$$

式中: $S_{-i}$ 为 $X_{-i}$ 的主效应值; $X_{-i}$ 为除 $X_i$ 以外所有输入变量。

根据式(9)可以将 $S_{-i}$ 写成:

$$S_{-i}=V_{-i}/V(Y)=\frac{V[E(Y/X_{-i})]}{V(Y)} \quad (15)$$

根据全方位方差公式:

$$V(Y)=V[E(Y/X_{-i})]+E[V(Y/X_{-i})] \quad (16)$$

可以将 $X_i$ 的总灵敏度 $S_{Ti}$ 写为

$$S_{Ti}=1-\frac{V[E(Y/X_{-i})]}{V(Y)}=\frac{E[V(Y/X_{-i})]}{V(Y)} \quad (17)$$

Sobol方法不仅可以计算出单个输入变量对函数的一阶灵敏度指标,还可以获得该变量对函数的总贡献。同时其计算精度较高,应用较为广泛。

### 3 MOEA/D-DEAH算法

MOEA/D-DE算法的性能在MOEA/D的基础上有了一定提升<sup>[8]</sup>,但是存在DE差分进化算子中超参数敏感的问题,不同问题对应的超参数设置有所不同。为了使超参数可以进行自适应调节,本文提出超参数具有自适应能力的MOEA/D-DE算法——MOEA/D-DEAH。

该算法首先通过拉丁超立方取样法对初始种群个体进行取样,然后根据初始种群个体位置与函数值构建Kriging代理模型,在获得代理模型后通过Sobol方法调用代理模型进行灵敏度分析来获得每个输入变量对不同目标的总灵敏度指标 $S_{Ti}$ 。MOEA/D-DE算法是通过将多目标问题分解成多个单目标问题,构成不同的单目标问题的目标权重不同,因此变量对不同单目标的灵敏度也不相同。本文将交叉控制参数 $C_R$ 从常数扩充为 $N \times D$ 大小的矩阵,其中 $N$ 为种群个体数量,同时

也是单目标问题的个数,  $D$  为问题的自变量维度。 $C_R$  的自适应公式为

$$C_{R_{n,i}} = 0.8 - 0.5 \times \exp\left(-20 \sum_{m=1}^M \tau_{n,m} S_{Ti,m}^T\right) \quad (18)$$

( $n = 1, \dots, N; i = 1, \dots, D; m = 1, \dots, M$ )

式中:  $n$  为第  $n$  个单目标问题;  $i$  为第  $i$  维自变量;  $m$  为第  $m$  个目标;  $\tau$  为目标权重。

通过式(18)可以使不同权重子问题不同变量得到其自己的交叉控制参数  $C_R$ 。

比例因子  $F$  是一个常数, 用来控制变异的程度。较小的  $F$  会使算法变异不足, 对优化空间的探索不足, 从而导致陷入局部最优; 而较大的  $F$  可以提高全局优化能力, 但也伴随着收敛速度的降低。因此,  $F$  的选取也十分重要。

Levy 飞行是一种随机游走的现象, 利用对称的 Levy 稳定分布, 用 Mantegna 方法<sup>[13]</sup> 生成随机步长, 如式(19)所示。

$$s = \frac{u}{|v|^{1/\beta}} \quad (19)$$

式中:  $u \sim N(0, \sigma^2); v \sim N(0, 1)$ 。

$$\sigma = \left\{ \frac{\Gamma(1+\beta) \sin(\pi\beta/2)}{\beta\Gamma[(1+\beta)/2] \cdot 2^{\beta-1/2}} \right\}^{1/\beta} \quad (20)$$

式中:  $\beta \in [1, 3]$ , 本文  $\beta$  取 1.5。

本文使用  $s$  来调节比例因子  $F$ :

$$FT_i = 0.4 \times s \quad (21)$$

式中:  $FT_i$  为比例因子  $F$  的临时取值; 下标  $i$  为第  $i$  个变量。

由于 Levy 飞行会产生极大的步长或者极小的步长, 通过式(22)来对  $F$  进行限制。

$$F_i = \begin{cases} 0.3 + 0.1\text{rand} & (FT_i < 0.3) \\ FT_i & (0.3 \leq FT_i \leq 2) \\ 2 - 0.1\text{rand} & (FT_i > 2) \end{cases} \quad (22)$$

式中: rand 为均匀分布在  $[0, 1]$  区间的随机数。

MOEA/D-DEAH 算法流程如下:

1) 输入: 优化问题, 停止条件, 邻居数量  $T$ , 种群大小  $N$ , 更新个数  $n_r$ , 邻居选择概率  $\delta$ ;

2) 初始化生成  $N$  个均匀分布的权重系数向量  $w(i), i = 1, \dots, N$ ;

3) 计算任意两个权重向量之间的欧氏距离,

对前  $T$  个离当前权重向量  $w(i)$  最近的权重向量记为  $B(i) = [i_1, \dots, i_T], i = 1, \dots, N$ ;

4) 在设计空间内使用拉丁超立方进行随机取样得到大小为  $N$  的种群  $X'$ , 并计算其相应的目标函数集  $Y'$ ;

5) 使用种群  $X'$  和其函数集  $Y'$  构建代理模型;

6) 使用 Sobol 灵敏度分析方法利用代理模型获取总灵敏度指标  $S_{Ti}$ ;

7) 用初始种群初始化向量  $z = [z_1, \dots, z_m]^T$ ;

8) 开始循环, 根据概率  $\delta$  选择邻居种群  $P(i)$  是  $B(i)$  还是整个种群, 从  $P(i)$  中随机选两个个体, 使用式(18)~式(20)改进的变异算子生成  $x_{\text{new}}$ , 并计算函数值  $f(x_{\text{new}})$ ;

9) 更新  $z$ : 如果  $f_j(x_{\text{new}}) < z_j$ , 则  $z_j = f_j(x_{\text{new}})$ ,  $j = 1, \dots, m$ ;

10) 更新种群: 对于  $P(i)$  中的单目标优化子问题, 使用相应的分解方法计算  $f(x_{\text{new}})$  对每个子问题的目标函数值  $g[x|w(j)], j \in B(i)$ , 确定邻居种群中所有  $g[x_{\text{new}}|w(j)] < g[x_j|w(j)]$  的个体编号, 并挑选随机  $n_r$  个个体进行更新;

11) 如果达到停止条件, 则停止算法并输出种群  $P_i$ ; 否则, 令  $t = t + 1$ , 转到步骤 8)。

## 4 函数测试

使用测试函数对 MOEA/D-DE $_{C_R=0.2}$  算法、MOEA/D-DE $_{C_R=0.5}$  算法、MOEA/D-DE $_{C_R=1.0}$  算法、NSGA II 算法和 MOEA/D-DEAH 算法的性能进行对比测试, MOEA/D 类算法分解方法均使用 Tchebycheff 分解方法。

测试函数主要有两组, 分别是由 Zhang Q 等<sup>[14]</sup> 提出的 UF 系列函数和 S. Huband 等<sup>[15]</sup> 提出的 WFG 系列函数。UF 系列函数的特点为设计自变量之间耦合较强, 问题较困难; WFG 系列函数包含了欺骗性问题, 混合形状 Pareto 前缘问题等较难的问题。上述函数的具体数学形式及其特点详见文献[14-15]。测试函数信息如表 1 所示, 表中给出测试函数时不同函数选择的自变量数量  $d$ , 目标数量  $m$ , 种群个体  $N$  以及迭代轮数  $G_{\text{max}}$ 。

表1 测试函数信息  
Table 1 Test function information

函数	$d$	$m$	$N$	$G_{\max}$
UF1	30	2	300	500
UF2	30	2	300	500
UF3	30	2	300	500
UF4	30	2	300	500
UF5	30	2	300	500
UF6	30	2	300	500
UF7	30	2	300	500
UF8	30	3	595	500
UF9	30	3	595	500
UF10	30	3	595	500
WFG1	14	2	100	250
WFG2	14	2	100	250
WFG3	14	2	100	250
WFG4	14	2	100	250
WFG5	14	2	100	250
WFG6	14	2	100	250

本文使用反世代距离(Inverted Generational Distance,简称IGD)指标作为评价算法得到的近似Pareto前缘的标准。IGD表达式为

$$I_{GD}(V, A) = \frac{\sum_{v \in V} d(v, A)}{|V|} \quad (23)$$

式中: $A$ 为算法得到的近似Pareto前缘; $V$ 为从真实Pareto前缘上提取的均匀分布且覆盖整个真实Pareto前缘的解的集合; $d(v, A)$ 为 $v$ 与 $A$ 中所有点的距离的最小值; $|V|$ 为 $V$ 中解的个数。

当集合 $A$ 较靠近真实Pareto前缘,并且分布均匀,可以获得较小的IGD。

不同超参数算法与MOEA/D-DEAH对每个测试函数30次独立运算所得最终IGD的均值和方差如表2所示,表中函数的指标最小值已被加粗,可以看出:MOEA/D-DEAH算法在大多数测试函数中取得了最好的结果,在其余测试函数中取得了第二名的成绩。

表2 同类算法测试函数算法IGD值对比  
Table 2 Comparison of IGD values of test function algorithms of similar algorithms

函数	MOEA/D-DE $C_R=0.2$		MOEA/D-DE $C_R=0.5$		MOEA/D-DE $C_R=1.0$		MOEA/D-DEAH	
	均值	方差	均值	方差	均值	方差	均值	方差
UF1	6.45E-02	4.24E-02	2.87E-02	2.98E-02	<b>3.29E-03</b>	<b>7.41E-04</b>	1.91E-02	4.27E-03
UF2	1.82E-02	1.49E-02	1.54E-02	8.55E-03	1.16E-02	2.97E-03	<b>1.12E-02</b>	<b>2.12E-03</b>
UF3	1.96E-01	4.37E-02	1.06E-01	3.49E-02	1.85E-02	<b>1.76E-02</b>	<b>1.04E-01</b>	3.29E-02
UF4	<b>4.37E-02</b>	<b>1.87E-03</b>	5.12E-02	3.12E-03	6.58E-02	5.72E-03	4.77E-02	2.03E-03
UF5	3.45E-01	1.10E-01	2.85E-01	<b>6.17E-02</b>	4.13E-01	9.50E-02	<b>2.24E-01</b>	1.02E-01
UF6	2.45E-01	1.86E-01	1.64E-01	<b>9.08E-02</b>	2.08E-01	1.83E-01	<b>1.46E-01</b>	9.78E-02
UF7	7.29E-02	1.28E-01	1.22E-02	5.04E-03	5.69E-03	1.85E-03	<b>1.11E-02</b>	<b>1.57E-03</b>
UF8	7.90E-02	3.19E-02	7.86E-02	5.07E-03	<b>7.37E-02</b>	1.28E-02	7.85E-02	<b>2.45E-03</b>
UF9	1.01E-01	6.58E-02	1.09E-01	6.74E-02	1.02E-01	7.17E-02	<b>8.36E-02</b>	<b>5.43E-02</b>
UF10	<b>2.90E-01</b>	9.75E-02	3.65E-01	7.14E-02	6.61E-01	8.38E-02	3.59E-01	<b>6.26E-02</b>
WFG1	<b>1.69E-02</b>	1.06E-03	1.73E-02	1.03E-03	1.77E-02	1.36E-03	<b>1.69E-02</b>	1.02E-03
WFG2	6.51E-02	3.90E-02	3.94E-02	1.59E-02	4.01E-02	5.01E-03	<b>3.68E-02</b>	<b>4.12E-03</b>
WFG3	1.78E-02	1.27E-03	<b>1.60E-02</b>	<b>5.89E-04</b>	1.74E-02	1.38E-03	1.62E-02	6.81E-04
WFG4	<b>2.14E-02</b>	<b>2.26E-03</b>	2.77E-02	2.95E-03	5.06E-02	6.61E-03	2.56E-02	4.18E-03
WFG5	7.00E-02	<b>3.01E-04</b>	6.99E-02	1.78E-04	7.01E-02	1.30E-04	<b>6.98E-02</b>	1.09E-03
WFG6	1.10E-01	<b>4.01E-02</b>	9.71E-02	4.74E-02	9.70E-02	5.96E-02	<b>8.56E-02</b>	6.17E-02

MOEA/D-DEAH算法和NSGA II算法对每个测试函数进行30次独立运算所得最终IGD的均值和方差如表3所示,表中函数的指标最小值已被

加粗,可以看出:MOEA/D-DEAH算法在大多数测试函数中领先于NSGA II算法,新算法不仅优于不同超参数的同类算法,还优于其他类型的经典算法。

表3 经典算法测试函数算法IGD值对比  
Table 3 Comparison of IGD values of classical algorithm test function algorithm

函数	MOEA/D-DEAH		NSGA II	
	均值	方差	均值	方差
UF1	1.91E-02	4.27E-03	9.76E-02	1.14E-02
UF2	1.12E-02	2.12E-03	3.99E-02	7.74E-03
UF3	1.04E-01	3.29E-02	1.98E-01	5.17E-02
UF4	4.77E-02	2.03E-03	4.34E-02	6.30E-04
UF5	2.24E-01	1.02E-01	2.33E-01	5.26E-02
UF6	1.46E-01	9.78E-02	1.26E-01	6.94E-02
UF7	1.11E-02	1.57E-03	6.85E-02	8.67E-02
UF8	7.85E-02	2.45E-03	2.67E-01	8.43E-03
UF9	8.36E-02	5.43E-02	1.84E-01	8.00E-02
UF10	3.59E-01	6.26E-02	3.48E-01	2.53E-02
WFG1	1.69E-02	1.02E-03	1.61E-02	2.36E-03
WFG2	3.68E-02	4.12E-03	1.58E-02	8.54E-04
WFG3	1.62E-02	6.81E-04	1.79E-02	1.53E-03
WFG4	2.56E-02	4.18E-03	1.77E-02	9.62E-04
WFG5	6.98E-02	1.09E-03	7.15E-02	9.20E-04
WFG6	8.56E-02	6.17E-02	8.79E-02	2.04E-02

综上所述,MOEA/D-DEAH算法的超参数自适应功能十分有效,可以根据不同的问题自动调整超参数的设置来得到较好的优化结果;同时,新算法具有较好的鲁棒性,验证了通过灵敏度分析提前提取问题相关信息来指导优化的可行性。

## 5 翼型气动隐身优化设计

### 5.1 翼型CFD算例验证

CFD二维算例验证采用经典算例NACA0012跨声速流场数值模拟,该算例被广泛应用于二维CFD计算校核。翼型的计算网格如图1所示,网格为C型拓扑,网格量为7.4万,远场距离翼型表面40倍弦长。

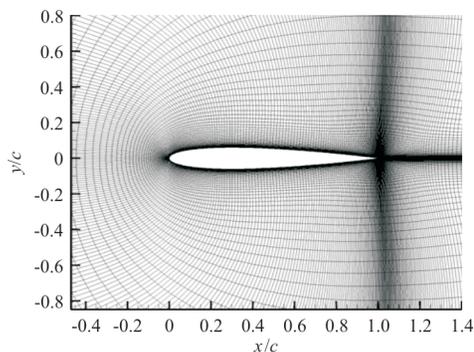


图1 NACA0012计算网格

Fig. 1 NACA0012 computing grid

二维翼型算例选取设计状态为自由来流马赫数  $Ma = 0.799$ , 雷诺数  $Re = 9 \times 10^6$ , 迎角  $\alpha = 2.26^\circ$ 。在CFD计算后与实验数据结果进行对比。计算时湍流模型选用  $k-\omega$  SST 湍流模型。数值模拟结果与实验值的压力分布对比如图2所示,可以看出:本文的数值模拟结果与实验值十分接近,可以准确地捕捉激波的位置和强度,表明本文采用的CFD数值计算方法在求解二维翼型的气动特性中具有较高的数值模拟精度。

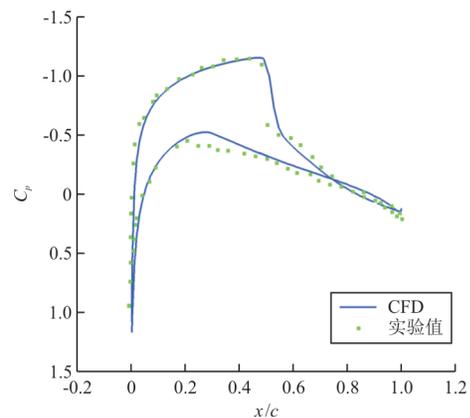


图2 翼型压力分布对比

Fig. 2 Comparison of airfoil pressure distribution

### 5.2 翼型隐身特性算例验证

使用矩量法对二维翼型的隐身特性进行计算,其中矩量法的二维计算算例验证采用NACA0012翼型的双站RCS算例,使用矩量法计算RCS并与文献[16-17]中的实验数据进行对比,来检验本文所使用的二维矩量法的计算精度。计算条件为TM波(Transverse Magnetic Waves)以方位角为入射,接收方位角为  $0 \sim 360^\circ$ ,翼型弦长为  $L = 10\lambda$ 。TM波垂直入射NACA0012翼型的方位角示意图如图3所示。NACA0012翼型双站RCS仿真与实验的对比如图4所示,可以看出:本文RCS的计算值和参考文献值基本吻合,表明本文所使用的矩量法在求解二维翼型的隐身特性中具有较高的数值模拟精度。

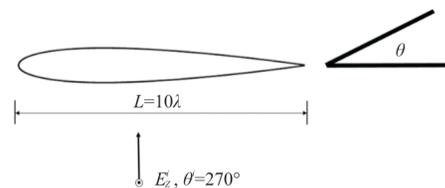


图3 TM波垂直入射NACA0012翼型的方位角示意图

Fig. 3 Azimuth diagram of a NACA0012 airfoil with a vertical incident TM wave

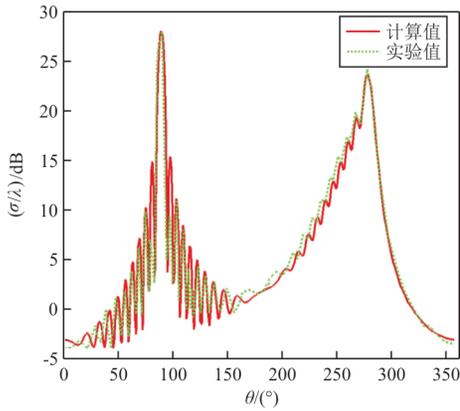


图 4 NACA0012翼型双站 RCS 计算值与实验值的对比  
Fig. 4 Comparison of calculated and experimental RCS values for NACA0012 airfoil

### 5.3 优化设计

优化算例为某机翼某站位的初始翼型,厚度要求为 14%。优化设计的初始翼型为钝后缘的 NACA65A014 翼型,设计状态为  $M_\infty = 0.75$ ,  $Re = 6.8 \times 10^6$ 。巡航状态下翼型在保持一定升力系数的情况下应有优秀的低阻、低可探测特性。隐身方面使用 TE 波对翼型前向俯仰  $\pm 30^\circ$  范围的 RCS 均值进行优化设计,频率选取 1 GHz。优化同时还应考虑结构布置等问题,如使翼型最大厚度不小于初始最大厚度。故优化目标及约束如下:

$$\begin{cases} \min \overline{S_{RCS(-30^\circ \sim 30^\circ)}} \\ \min C_d \\ C_L = 0.4523 \\ C_m \geq -0.01824 \\ t_{\max} \geq 0.14 \\ C_D \leq 0.02625 \\ \overline{S_{RCS(-30^\circ \sim 30^\circ)}} \leq 0.0562 \end{cases} \quad (24)$$

式中:  $C_L$  为升力系数;  $C_m$  为力矩系数;  $C_D$  为阻力系数;  $t_{\max}$  为翼型最大厚度;  $\overline{S_{RCS(-30^\circ \sim 30^\circ)}}$  为前向  $\pm 30^\circ$  范围的 RCS 均值。

翼型参数化使用扰动 CST 方法<sup>[18-19]</sup>,设计变量为 18 个,翼型上下表面各 9 个。分别采用 MOEA/D-DE $_{C_r=0.2}$  算法、MOEA/D-DE $_{C_r=0.5}$  算法、MOEA/D-DE $_{C_r=1.0}$  算法、NSGA II 算法和 MOEA/D-DEAH 算法进行翼型优化设计。算法设置中种群个体  $N = 100$ ,迭代轮数  $G_{\max} = 100$ 。算法收敛判断使用超体积算子  $H_V$ <sup>[20]</sup>来判断。

超体积算子  $H_V$  是评价多目标优化性能指标

之一,表示由当前 Pareto 解集中个体与参考点在目标中围成的超立方体的体积。本文参考点选取初始翼型的阻力系数和 RCS 均值。

超体积收敛历程如图 5 所示,可以看出:新算法在初期收敛速度极快,在 20 步之后与新算法初期速度类似的 MOEA/D-DE $_{C_r=0.2}$  收敛速度放缓;20 步~40 步之间新算法与 MOEA/D-DE $_{C_r=0.5}$  超体积较为相近,但是随着迭代进行,新算法超体积与 MOEA/D-DE $_{C_r=0.5}$  拉开了差距;NSGA II 算法在 30 轮之间具有较高的  $H_V$  但随着迭代进行收敛性逐渐不足;同时,优化末期,MOEA/D-DEAH 算法在收敛性和收敛速度上均好于其他算法,算法差距有增大的趋势。

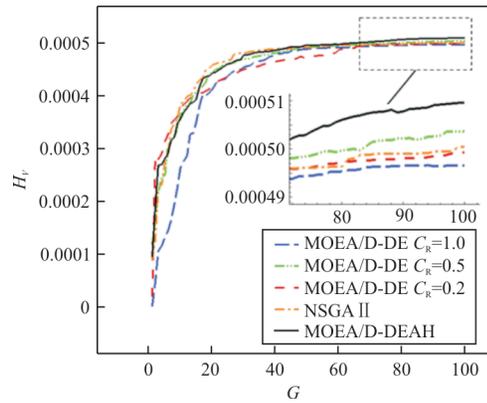


图 5 超体积迭代收敛图  
Fig. 5 Hypervolume iterative convergence diagram

不同算法优化后得到的结果如图 6 所示,前缘解中间段的细节和被选展示点的位置如图 7 所示,被选展示点翼型对比如图 8 所示。被选展示点性能如表 4 所示,表中“T”为厚度。

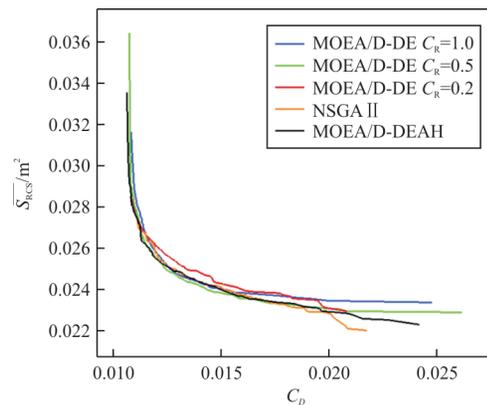


图 6 算法结果对比图  
Fig. 6 Comparison of algorithm results

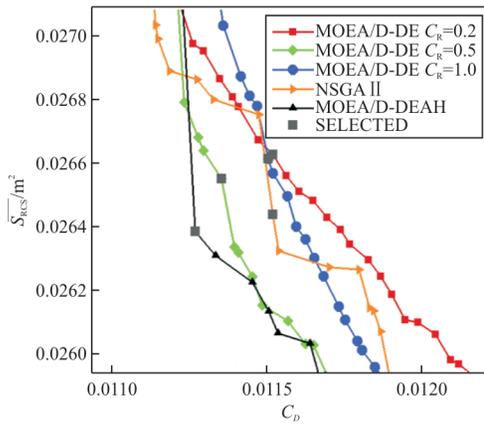


图 7 Pareto前缘中间细节及被选展示点位置  
Fig. 7 Middle details of Pareto leading edge and location of selected display point

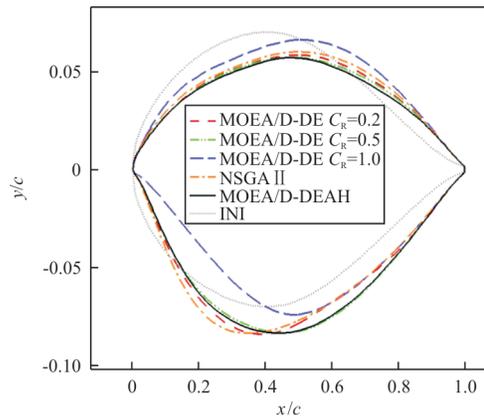


图 8 被选展示点翼型对比图  
Fig. 8 Selected to show point airfoil comparison

表 4 被选展示点性能  
Table 4 Selected point performance

对比项	性能			
	$C_D$	$\overline{S_{RCS}}/m^2$	$C_m$	$T/c$
初始翼型	0.026 25	0.056 20	-0.018 24	0.140 0
MOEA/D-DE $_{C_r=0.2}$	0.011 52	0.026 63	0.032 87	0.140 5
MOEA/D-DE $_{C_r=0.5}$	0.011 35	0.026 55	0.047 14	0.140 4
MOEA/D-DE $_{C_r=1.0}$	0.011 50	0.026 61	0.023 66	0.140 0
NSGA II	0.011 51	0.026 44	0.021 96	0.140 3
MOEA/D-DEAH	0.011 27	0.026 38	0.043 48	0.140 2

从图 8 可以看出:MOEA/D-DEAH 算法与 MOEA/D-DE $_{C_r=0.5}$  算法得到的翼型较为相近。

从表 4 可以看出:MOEA/D-DEAH 算法与 MOEA/D-DE $_{C_r=0.5}$  算法的性能参数比较接近,二者的前缘半径较初始翼型(INI)均有较大幅度的减小,上表面厚度降低,同时更为平坦,下表面厚度增加;MOEA/D-DE $_{C_r=0.2}$  算法得到的翼型与上述两者相比,上表面厚度有小幅增加,下表面厚度相

近,但是最厚点略有前移;MOEA/D-DE $_{C_r=1.0}$  算法得到的翼型与上述三者差别较大,与 MOEA/D-DEAH 算法结果相比上表面厚度有大幅增加,下表面前部后缩幅度较大,厚度有大幅减少,其性能与 MOEA/D-DE $_{C_r=0.2}$  算法结果相近;NSGA II 算法优化获得的翼型上表面高度为除了 MOEA/D-DE $_{C_r=1.0}$  算法得到的结果外最高的,下表面与除了 MOEA/D-DE $_{C_r=1.0}$  算法得到的结果相比相差较大,翼型下表面最低点最为靠前。

被选展示点的压力系数和隐身特性分别如图 9~图 10 所示。

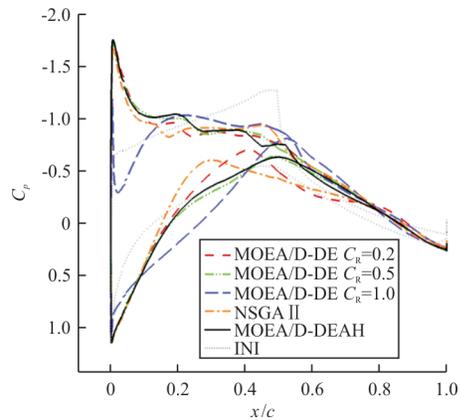


图 9 被选展示点压力系数对比图  
Fig. 9 The point pressure coefficient comparison diagram was selected to be displayed

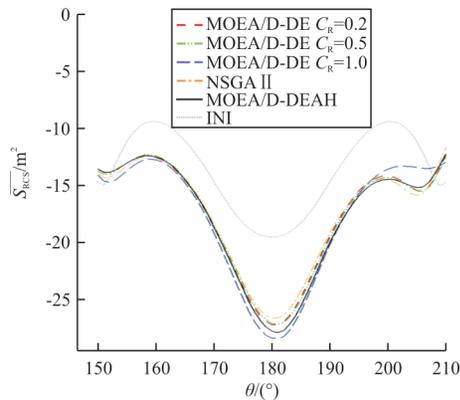


图 10 被选展示点隐身特性对比图  
Fig. 10 Contrast diagram of stealth characteristics of selected display points

从图 9 可以看出:各优化算法结果都较初始结果有了较大提升,其中 MOEA/D-DE $_{C_r=1.0}$  得到的结果与其他算法所得优化结果差异较大,首先是前缘的激波更强,但是吸力并没有其他高;其次是翼型为了弥补后部缺失的升力,中前部产生的升力更多;NSGA II 与 MOEA/D-DE $_{C_r=0.2}$  得到的结

果在下表面与MOEA/D-DE $c_r=0.5$ 和MOEA/D-DEAH算法较为不同,其最大吸力产生处更靠前,这也符合翼型对比分析的结果;MOEA/D-DEAH算法得到的结果最为优秀,其上表面除前缘激波外整体较为平稳。

从图10可以看出:优化后的翼型其隐身性能较初始外形都有很大提升,在优化结果中,MOEA/D-DE $c_r=1.0$ 算法得到的结果在较大角度范围内隐身都较为出色,其次是MOEA/D-DEAH算法得到的结果;但是MOEA/D-DE $c_r=1.0$ 算法得到的结果在 $200^\circ\sim 210^\circ$ 较差,因为本文在取平均值时并未使用dBsm,这导致MOEA/D-DE $c_r=1.0$ 算法在较大范围的优势被 $200^\circ\sim 210^\circ$ 的劣势抵消,使得表4中MOEA/D-DE $c_r=1.0$ 算法的RCS均值比MOEA/D-DEAH算法要差;NSGA II结果在偏离头向时表现较好,但是在头向表现较差,最终获得了仅次于MOEA/D-DEAH算法的RCS均值;而MOEA/D-DEAH算法得到的结果在所有角度都有着非常稳定的表现,因此在均值上得到的结果最好。

综上所述,MOEA/D-DEAH算法在翼型气动隐身多目标优化中通过自动调节超参数在收敛性和分布性上均可获得良好的效果。在超体积收敛方面,新算法无论在前期收敛速度、后期收敛速度和超体积大小等方面均好于其他算法;新算法在近似Pareto前缘对比中具有较强的竞争力,计算得到的近似Pareto前缘的大多数区域都支配了其他算法产生的近似Pareto前缘,体现了本文提出的超参数自适应机制的有效性与新算法的鲁棒性。在翼型气动隐身多目标优化问题中超参数自适应机制可以将算法超参数调整至较好状态,从而获得综合性能优良的结果。

## 6 结 论

1) 为了解决超参数敏感问题,本文提出了超参数自适应的MOEA/D-DE算法——MOEA/D-DEAH,在函数测试中新算法展现出较好的鲁棒性,可以在大多数函数中取得较好结果。

2) 在翼型气动隐身优化问题中,MOEA/D-DEAH算法在超体积指标上好于其他算法,获得了较好的近似Pareto前缘。函数测试和翼型气动隐身优化设计验证了MOEA/D-DEAH算法的超参数自适应能力和鲁棒性。

目前大多数算法仍不可避免超参数敏感的问题,如何在保证效率的前提下更好地提取问题的相关信息,并将其应用在参数调节上,将会是今后重要的研究方向。

## 参 考 文 献

- [1] 夏露,张欣,杨梅花,等. 飞翼布局翼型气动隐身综合设计[J]. 西北工业大学学报, 2017, 35(5): 821-826.  
XIA Lu, ZHANG Xin, YANG Meihua, et al. Airfoil aerodynamic stealth integrated design for a flying wing configuration[J]. Journal of Northwestern Polytechnical University, 2017, 35(5): 821-826. (in Chinese)
- [2] 张彬乾,罗烈,陈真利,等. 飞翼布局隐身翼型优化设计[J]. 航空学报, 2014, 35(4): 957-967.  
ZHANG Binqian, LUO Lie, CHEN Zhenli, et al. On stealth airfoil optimization design for flying wing configuration[J]. Acta Aeronautica et Astronautica Sinica, 2014, 35(4): 957-967. (in Chinese)
- [3] DEB K, PRATAP A, AGARWAL S, et al. A fast and elitist multiobjective genetic algorithm: NSGA-II[J]. IEEE Transactions on Evolutionary Computation, 2002, 6(2): 182-197.
- [4] ZHANG Q, LI H. MOEA/D: a multiobjective evolutionary algorithm based on decomposition[J]. IEEE Transactions on Evolutionary Computation, 2007, 11(6): 712-731.
- [5] ZITZLER E, KÜNZLI S. Indicator-based selection in multiobjective search[C]// International Conference on Parallel Problem Solving from Nature. Heidelberg, Berlin: Springer, 2004: 832-842.
- [6] LIU Q, JIN Y, HEIDERICH M, et al. An adaptive reference vector-guided evolutionary algorithm using growing neural gas for many-objective optimization of irregular problems[J]. IEEE Transactions on Cybernetics, 2020, 159: 1-14.
- [7] JIANG S, YANG S, MEMBER S, et al. Scalarizing functions in decomposition-based multiobjective evolutionary algorithms[J]. IEEE Transactions on Evolutionary Computation, 2018, 22(2): 296-313.
- [8] LI H, ZHANG Q. Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II[J]. IEEE TEVC, 2009, 13(2): 284-302.
- [9] ZHU X, GAO Z, DU Y, et al. A decomposition-based multi-objective optimization approach considering multiple preferences with robust performance[J]. Applied Soft Computing, 2018, 73: 263-282.
- [10] DAS S, SUGANTHAN P N. Differential evolution: a survey of the state-of-the-art[J]. IEEE Transactions on Evolutionary Computation, 2010, 15(1): 4-31.
- [11] 朱新奇. 飞行器外形优化设计方法及多输出代理模型研究[D]. 西安: 西北工业大学, 2019.  
ZHU Xinqi. Research on flight vehicle shape optimization design algorithms and multi-output surrogate model[D]. Xi'an: Northwestern Polytechnical University, 2019. (in Chinese)

