

文章编号: 1674-8190(XXXX)XX-001-11

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 CR 和 F 对算法性能影响较大。基于 MOEA/D-DE 算法框架, 利用 Sobol 全局灵敏性分析方法对差分进化算子中的交叉控制参数 CR 进行改进, 使用莱维飞行策略控制比例因子 F, 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 NSGAI 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; 超参数; 灵敏性分析; 气动隐身优化; 差分进化算子

中图分类号: V221.3

文献标识码: A

DOI: 10.16615/j.cnki.1674-8190.XXXX.XX.01

开放科学(资源服务)标识码(OSID):



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 CR and F 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 CR in the differential evolution operator, and the Levy flight strategy is used to control the scale factor F. 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 NSGAI 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; 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资助(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, XXXX, XX(XX): 1-11.

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, XXXX, XX(XX): 1-11. (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^[1]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^[2]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NSGAI(Non-dominated Sorting Genetic Algorithm II)^[3]、基于分解的多目标优化算法MOEA/D(A Multiobjective Evolutionary Algorithm Based on Decomposition)^[4]、基于指标的多目标优化算法IBEA(Indicator-based Evolutionary Algorithm)^[5]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,性能比较优秀。目前,MOEA/D算法以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^[6]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到真实Pareto前缘,性能较好;S. Jiang等^[7]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(an 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。这个方向上国内外研究较少;Li H等^[8]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,MOEA/D-DE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,其对以下两个主要控制参数敏感:交叉控制参数CR和比例因子F。交叉控制参数CR的取值与Pareto解集的复杂度呈现正相关^[9-10],比例因子F则控制了算法的局部搜索能力与全局搜索能力。这些参数的选择往往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,给处理工程优化问题造成了一定障碍。朱新奇^[11]提出了局部坐标交叉算子改善的MOEA/D-DE算法,减小了对CR值的依赖,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^[12]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灵敏度分析来获得多目标问题中每个自变量对于不同问题的灵敏度,然后根据自变量灵敏度越高对问题越重要、需要进行变异的次数越多的原则对交叉控制参数CR进行修改,使CR值根据自变量不同与子目标权重不同进行自适应。再利用莱维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。通过以上操作获得自适应超参数的MOEA/D-DE算法——MOEA/D-DEAH(A 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^[4]用分解方法将多目标问题分解为若干个子优化问题,再用EA(进化算法)来同时优化这些子问题,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,所以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,根据第一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,否则返回第二步。

本文中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

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

$$\text{s.t. } x \in \Omega$$

其中 $w = [w_1, \dots, w_m]^T$ 为权重系数向量,满足 $w_i \geq 0$, 且 $\sum_{i=1}^m w_i = 1$. $z^* = [z_1^*, \dots, z_m^*]^T$ 为理想参考点 (ideal reference point), 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。

1.2 MOEA/D-DE 算法

文献[8]指出,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,并提出了使用 DE 算子后再使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

$$v_i = x_{r1} + F \cdot (x_{r2} - x_{r3}) \quad (i \in 1, 2, \dots, NP) \quad (2)$$

其中 NP 表示种群大小,和 i 不同、 $r1, r2, r3$ 是在 $[1, NP]$ 中随机抽取的不同的整数, F 是比例因子。 v 为突变向量, x 为个体向量。

交叉操作结合突变向量 v_i 和目标向量 x_i , 生成试验向量 u_i , 如下式所示

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

其中 j_{rand} 是介于 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, $\text{rand}_j(0, 1)$ 是 0 到 1 之间均匀分布的随机数, CR 是交叉控制参数。

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。多项式变异算子如下

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

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

式(4)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, $p_m = 1/D$ 。式(5)中 μ 为 0 到 1 之间均匀分布的随机数, η_m 为变异参数取值为 20。

另一方面,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,由此可以提高扩大搜索范围,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 限制更新的个数,从而提高种群的多样性。

但是 MOEA/D-DE 算法跟其他算法一样,算法性能也受超参数的影响。为了解决这个问题,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 Sobieski 在 1990 年提出的一种基于 Monte Carlo 积分灵敏度分析方法^[12],其通过方差分解 (Analysis of Variance, ANOVA) 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下式

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

其中, $\mathbf{X} = (X_1, X_2, \dots, X_n)$ 表示 n 维输入变量, Y 表示一维输出变量, 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:

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

由上式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(5)右端各项互相正交, 因此对式(5)左右两端求方差可得下式:

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

其中:

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

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 $V(Y)$, 可以得到下式:

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

上式中 S_i 表示的是单个变量 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} 表示的是变量 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, 其定义如下式所示:

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

其中 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} 写成下式

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

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

MOEA/D-DE^[8]算法性能在 MOEA/D 的基础上有了一定的提升, 但是面临 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, 本文提出了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性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, 所以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 CR 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。 CR 的自适应公式如下式所示

$$\begin{aligned} CR_{n,i} &= 0.8 - 0.5 \times \exp(-20 \sum_{m=1}^M \omega_{n,m} S_{Ti,m}^T) \\ n &= 1, \dots, N \\ i &= 1, \dots, D \\ m &= 1, \dots, M \end{aligned} \quad (18)$$

其中 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, ω 为目标权重。通过上式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 CR 。

比例因子 F 是一个常数, 用来控制变异的程度。较小的 F 会导致算法变异不足, 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 可以提高了全局优化的能力, 但是也伴随着收敛速度的降低, 所以缩放因子的选取也十分重要。

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^[13]生成随机步长, 公式如下式所示:

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

其中 $u \sim N(0, \sigma^2)$, $v \sim N(0, 1)$, σ 如下式所示:

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

式中 $\beta \in [1, 3]$, 本文 β 取 1.5。本文使用 $step$ 来调节比例因子 F , 如下式所示:

$$FT_i = 0.4 * step \quad (21)$$

上式中 FT_i 为比例因子的 F 的临时取值, 角标 i 表示第 i 个变量。由于莱维飞行会产生极大的步长或者极小的步长所以通过下式来对 F 进行限制。

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

上式中 $rand$ 表示均匀分布在 $[0, 1]$ 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 nr , 邻居选择概率 δ 。

(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 $B(i) = \{i_1, \dots, i_T\}$, 其中 $i = 1, \dots, N$;

(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 灵敏性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 δ 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(16)~式(20)改进的变异算子式(3-11)生成 x_{new} , 并计算函数值 $f(x_{new})$;

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

(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 $f(x_{new})$ 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 $g(x_{new}|w(j)) < g(x_j|w(j))$ 的个体编号, 并挑选随机 nr 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_t ; 否则, 令 $t = t + 1$, 转到步骤 8。

4 函数测试

本节使用测试函数对 MOEA/D-DE_{CR=0.2} 算法, MOEA/D-DE_{CR=0.5} 算法, MOEA/D-DE_{CR=1} 算法, NSGAII 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, 它们分别是由文献[14]提出的 UF 系列函数和文献[15]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可参考相应文献。测试函数信息如表 1 所示, 表中给出了测试函数时的不同函数选择的自变量数量 d , 目标数量 m , 种群个体 N 个以及迭代轮数 G_{max} 。

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

function	d	m	N	Gmax
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 IGD (Inverted Generational Distance) 指标作为评价算法得到的近似 Pareto 前缘的标准。IGD 表达式为

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

其中 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 值。

不同超参数 MOEA/D-DE 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, 其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

Function	MOEA/D-DE _{CR=0.2}		MOEA/D-DE _{CR=0.5}		MOEA/D-DE _{CR=1}		MOEA/D-DEAH	
	mean	std	mean	std	mean	std	mean	std
UF1	6.45E-02	4.24E-02	2.87E-02	2.98E-02	3.29E-03	7.41E-04	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	1.12E-02	2.12E-03
UF3	1.96E-01	4.37E-02	1.06E-01	3.49E-02	1.85E-02	1.76E-02	1.04E-01	3.29E-02
UF4	4.37E-02	1.87E-03	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	6.17E-02	4.13E-01	9.50E-02	2.24E-01	1.02E-01
UF6	2.45E-01	1.86E-01	1.64E-01	9.08E-02	2.08E-01	1.83E-01	1.46E-01	9.78E-02
UF7	7.29E-02	1.28E-01	1.22E-02	5.04E-03	5.69E-03	1.85E-03	1.11E-02	1.57E-03
UF8	7.90E-02	3.19E-02	7.86E-02	5.07E-03	7.37E-02	1.28E-02	7.85E-02	2.45E-03
UF9	1.01E-01	6.58E-02	1.09E-01	6.74E-02	1.02E-01	7.17E-02	8.36E-02	5.43E-02
UF10	2.90E-01	9.75E-02	3.65E-01	7.14E-02	6.61E-01	8.38E-02	3.59E-01	6.26E-02
WFG1	1.69E-02	1.06E-03	1.73E-02	1.03E-03	1.77E-02	1.36E-03	1.69E-02	1.02E-03
WFG2	6.51E-02	3.90E-02	3.94E-02	1.59E-02	4.01E-02	5.01E-03	3.68E-02	4.12E-03
WFG3	1.78E-02	1.27E-03	1.60E-02	5.89E-04	1.74E-02	1.38E-03	1.62E-02	6.81E-04
WFG4	2.14E-02	2.26E-03	2.77E-02	2.95E-03	5.06E-02	6.61E-03	2.56E-02	4.18E-03
WFG5	7.00E-02	3.01E-04	6.99E-02	1.78E-04	7.01E-02	1.30E-04	6.98E-02	1.09E-03
WFG6	1.10E-01	4.01E-02	9.71E-02	4.74E-02	9.70E-02	5.96E-02	8.56E-02	6.17E-02

MOEA/D-DEAH 算法和 NSGAII 算法对每个测试函数 30 次独立运算所得最终 IGD 的均值和方差如表 3 所示, 其中函数的指标最小值已被加粗, 可以看出: MOEA/D-DEAH 算法在大多数函数中领先于 NSGAII 算法, 新算法不仅好于不同超参数的同类算法还优于其他类型的经典算法。

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。同时体现了新算法具有较好的鲁棒性, 重要的是验证了通过灵敏性分析提前提取问题相关信息来指导优化的可行性。

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

Function	MOEA/D-DEAH		NSGAI	
	mean	std	mean	std
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

5 翼型气动隐身优化设计

5.1 翼型 CFD 算例验证

本文 CFD 二维算例验证采用经典算例 NACA0012 跨音速流场数值模拟,该算例被广泛应用于二维 CFD 计算校核。该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。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。

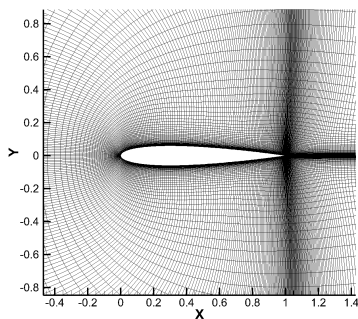


图1 NACA 0012 计算网格
Fig.1 NACA 0012 computing grid

本文二维翼型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:本文的数值模拟的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。这说明本文采用的 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。

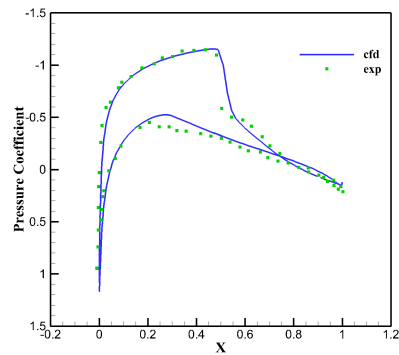


图2 翼型压力分布对比
Fig.2 Comparison of airfoil pressure distribution

5.2 翼型隐身特性算例验证

本文使用矩量法来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 NACA 0012 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 NACA 0012 翼型的方位角示意图如图 3 所示,NACA 0012 翼型双站 RCS 仿真与实验的对比图如图 4 所示,可以看出:本文 RCS 的计算值和参考文献值二者基本吻合,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。

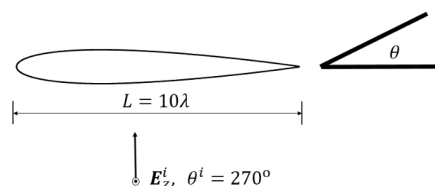


图3 TM 波垂直入射 NACA 0012 翼型的方位角示意图
Fig.3 Azimuth diagram of a NACA 0012 airfoil with a vertical incident TM wave

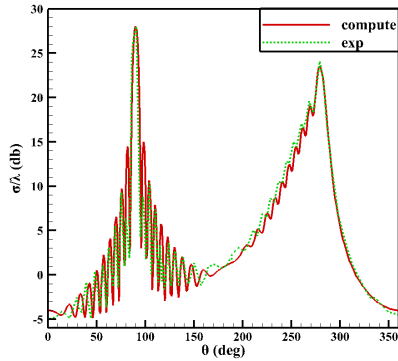


图4 NACA 0012翼型双站RCS计算值与实验值的对比
Fig. 4 Comparison of calculated and experimental RCS values for NACA 0012 airfoil

5.3 优化设计

本节优化算例为某机翼某站位的初始翼型,厚度要求为14%。故优化设计的初始翼型为修改为钝后缘的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1GHz。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

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

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; $\overline{RCS}_{(-30^\circ \sim 30^\circ)}$ 为前向 $\pm 30^\circ$ 范围的RCS均值。

翼型参数化使用扰动CST方法^[18-19],设计变量为18个,翼型上下表面各9个。分别采用MOEA/D-DE_{CR=0.2}算法,MOEA/D-DE_{CR=0.5}算法,MOEA/D-DE_{CR=1}算法,NSGAI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HV^[20]来判断。

超体积算子HV是评价多目标优化性能的指

标之一,它表示由当前Pareto解集中个体与参考点在目标中围成的超立方体的体积。本节中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_{CR=0.2}收敛速度放缓。20步至40步之间新算法与MOEA/D-DE_{CR=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_{CR=0.5}拉开了差距。NSGAI算法在30轮之间具有较高的HV但随着迭代进行收敛性逐渐不足。同时,优化末期,MOEA/D-DEAH算法在优化末期的收敛性上和收敛速度上均好于其他算法,算法差距有增大的趋势。

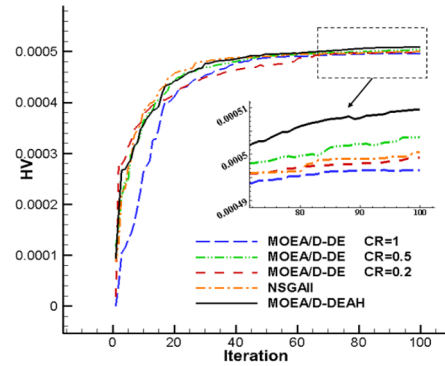


图5 超体积迭代收敛图

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图如图8所示。被选展示点性能如表4所示。

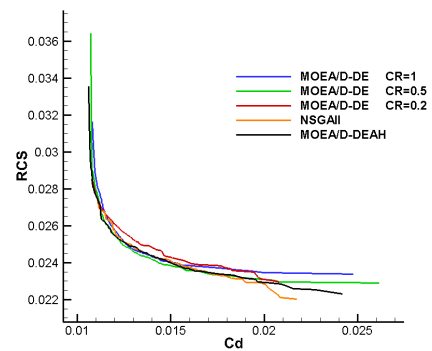


图6 算法结果对比图

Fig. 6 Comparison of algorithm results

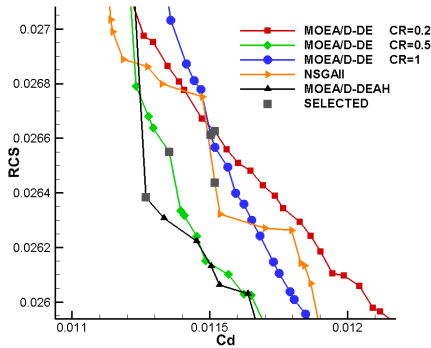


图7 Pareto 前缘中间细节及被选展示点位置
Fig. 7 Middle details of Pareto leading edge and location of selected display point

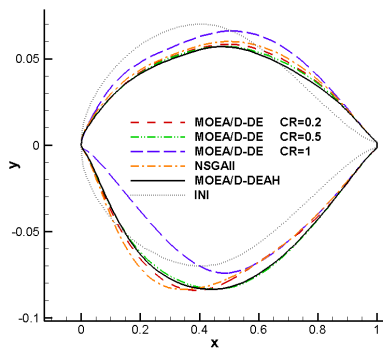


图8 被选展示点翼型对比图
Fig. 8 Selected to show point airfoil comparison

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

性能	C_D	RCS	C_M	厚度
初始翼型	0.026 25	0.056 20	-0.018 24	0.140 0
MOEA/D-DECR=0.2	0.011 52	0.026 63	0.032 87	0.140 5
MOEA/D-DECR=0.5	0.011 35	0.026 55	0.047 14	0.140 4
MOEA/D-DECR=1	0.011 50	0.026 61	0.023 66	0.140 0
NSGAII	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_{CR=0.5}算法得到的翼型较为相近。

从表4可以看出:两者的性能参数也比较接近。二者的前缘半径较初始翼型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_{CR=0.2}算法得到的翼型与前两者相比,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_{CR=1}算法得到的翼型与前三者差别较大,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_{CR=0.2}算法结果相近。经过NSGAII算法优化获得的翼型上表面高度为除了MOEA/D-DE_{CR=1}算法得到的结果外最高者,下表面与除了MOEA/D-DE_{CR=1}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。

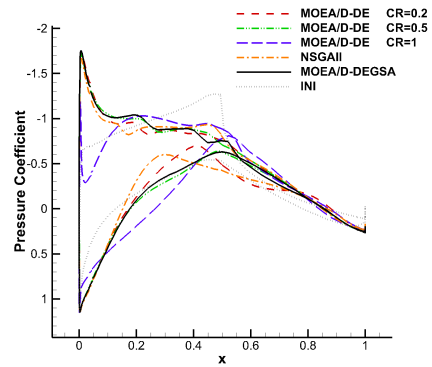


图9 被选展示点压力系数对比图
Fig. 9 The point pressure coefficient comparison diagram was selected to be displayed

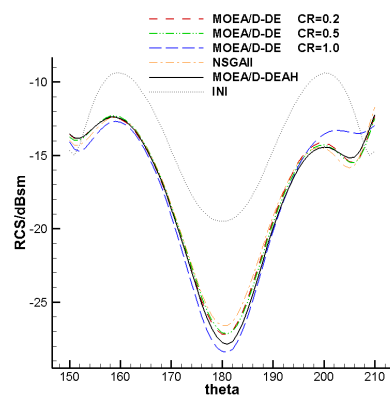


图10 被选展示点隐身特性对比图
Fig. 10 Contrast diagram of stealth characteristics of selected display points

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_{CR=1.0}得到的结果与其余几个优化结果差异较大,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。NSGAII与MOEA/D-DE_{CR=0.2}得到的结果在下表面与MOEA/D-DE_{CR=0.5}和MOEA/D-DEAH算法的较为不同,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:优化后的几种翼型隐身较初始外形都有很大提升,在几种优化结果中,MOEA/D-DE_{CR=1.0}算法得到的结果在较大范围角度隐身都较为出色,其次是 MOEA/D-DEAH 算法得到的结果;但是 MOEA/D-DE_{CR=1.0}算法得到的结果在 200°至 210°较差,因为本文在取平均值时并未使用 dBsm,这导致 MOEA/D-DE_{CR=1.0}算法在较大范围的优势被 200°至 210°的劣势抵消,使得表 4 中 MOEA/D-DE_{CR=1.0}算法的 RCS 均值比 MOEA/D-DEAH 算法的差。NSGAII 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,所以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以获得良好的效果。其中在超体积收敛方面,新算法无论在前期收敛速度快,后期收敛速度和超体积大小均好于其他算法;新算法在近似 Pareto 前缘对比中具有较强的竞争力,新算法计算得到的近似 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。这些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

本文为了解决超参数敏感问题,提出了超参数自适应的 MOEA/D-DE 算法——MOEA/D-DEAH。该算法基于灵敏度修改算子中的 CR,基于 Levy 飞行修改算子中的 F。为了验证算法有效性,进行了函数测试和翼型气动隐身优化设计。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;在翼型气动隐身优化问题中,新算法在超体积指标上好于其他几种算法,获得了较好近似 Pareto 前缘。函数测试和翼

型气动隐身优化问题验证了 MOEA/D-DEAH 算法的超参数自适应能力和鲁棒性。

目前大多数算法仍避免不了超参数敏感的问题,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 sur-

- vey 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)
- [12] SOBOL I M. Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates[J]. Mathematics and Computers in Simulation, 2001, 55 (1-3) : 271-280.
- [13] YANG X S. Nature-inspired metaheuristic algorithms[M]. Luniver Press, 2010.
- [14] ZHANG Q, ZHOU A, ZHAO S, et al. Multiobjective optimization test instances for the CEC 2009 special session and competition: Tech. Rep. CES-487 [R]. The School of Computer Science and Electronic Engineering, University of Essex, 2009.
- [15] HUBAND S, HINGSTON P, BARONE L, et al. A review of multiobjective test problems and a scalable test problem toolkit[J]. IEEE Transactions on Evolutionary Computation, 2006, 10(5): 477-506.
- [16] 许勇. 麦克斯韦方程组的时域有限体积法(FVTD)数值模拟[D]. 绵阳: 中国空气动力研究与发展中心, 2002.
- XU Yong. Numerical simulation of Maxwell equations by finite volume time domain method (FVTD)[D]. Mianyang: China Aerodynamics Research and Development Center, 2002. (in Chinese)
- [17] VINH H, DWYER H A, Van DAM C P. Finite-difference algorithms for the time-domain Maxwell's equations—a numerical approach to RCS analysis[C]. USA: AIAA-92-2989.
- [18] KULFAN B M. Universal parametric geometry representation method[J]. Journal of aircraft, 2008, 45(1): 142-158.
- [19] 卜月鹏, 宋文萍, 韩忠华, 等. 基于 CST 参数化方法的翼型气动优化设计[J]. 西北工业大学学报, 2013, 31(5): 829-836.
- BU Yuepeng, SONG Wenping, HAN Zhonghua, et al. Aerodynamic optimization design of airfoil based on CST parameterization method [J]. Journal of Northwestern Polytechnical University, 2013, 31(5): 829-836. (in Chinese)
- [20] AUGER A, BADER J, BROCKHOFF D, et al. Theory of the hypervolume indicator: optimal μ -distributions and the choice of the reference point[C]// The tenth ACM SIGEVO Workshop on Foundations of Genetic Algorithms. 2009: 87-102.

作者简介:

王培君(1998—),男,硕士研究生。主要研究方向:飞行器气动隐身一体化优化等。

夏露(1977—),女,博士,副教授。主要研究方向:飞行器设计,飞行器气动隐身一体化优化。

栾伟达(1998—),男,硕士研究生。主要研究方向:飞行器气动隐身一体化优化等。

陈会强(1997—),男,硕士研究生。主要研究方向:飞行器设计,电磁计算等。

(编辑:马文静)