

用于快速仿真优化的改进差分进化算法及其应用

饶大林, 蔡国飙

(北京航空航天大学宇航学院, 北京 100191)

摘 要: 提出一种改进的差分进化算法, 采用一种“位变”方法计算收缩因子, 该方法首先根据适应值对种群排序, 然后根据各个体的排列位置确定收缩因子; 采用正态分布函数对算法参数进行随机扰动来维持种群的多样性; 该算法还提出一种新的变异算子, 并将其与基本的差分变异算子结合使用以提高算法的寻优精度。经过对多个 Benchmark 函数的测试、分析和比较, 结果表明该算法具有较高的收敛精度和较快的收敛速度。最后将该算法用于火箭发动机涡轮气动优化, 以较小的计算成本将涡轮气动效率提高了 2.5%。应用结果表明该算法适用于快速仿真优化问题, 能有效地节约计算成本。

关键词: 差分进化; “位变”收缩因子; 正态分布; 变异算子; 仿真优化; 涡轮

中图分类号: V434 **文献标识码:** A **文章编号:** 1000-1328(2010)03-0793-05

DOI: 10.3873/j.issn.1000-1328.2010.03.028

0 引言

工程设计优化问题中, 数值仿真计算往往是非常耗时的, 比如流场或温度场的优化问题, 每一次仿真计算消耗的时间通常都要以小时甚至以天计算。因此工程仿真优化问题通常要求以较少的仿真次数获得满意的改进解, 这对算法的性能提出了较高的要求。

差分进化^[1](DE)是由 Storn 和 Price 等人提出的基于种群的随机搜索算法。DE 采用实数编码, 控制参数少, 寻优效率较高, 简单高效的优点使其在众多领域得到了广泛的应用。但 DE 仍然存在一些不足, 标准的 DE 采用“贪婪”选择策略, 虽然可以加快收敛速度, 但也会大大提高早熟收敛的概率; 后期的收敛速度较慢, 表现不够稳健; 标准的 DE 中, 收缩因子 F 和交叉概率 CR 都采用固定值, 而对于特定的优化对象, DE 的性能与参数的选取密切相关。为此, 近年来众多学者对 DE 进行了深入研究并进行了改进^[2]。比如文献[3]提出了随机搜索因子和“时变”收缩因子的改进差分进化算法, 文献[4]提出了三角变异算子。

本文首先在已出版文献的研究成果的基础上进一步研究, 提出一种改进的差分进化算法(MDE), 该

算法采用“位变”收缩因子和“时变”交叉概率; 用正态分布函数对算法参数进行随机扰动以增强种群的多样性; 还提出一种新的变异算子, 并将此算子与基本的差分变异算子相结合以提高算法的收敛精度; 然后对算法进行测试和比较分析; 最后将其应用于火箭发动机的涡轮叶片气动优化问题。

1 基本的差分进化算法

基本的 DE 由三个操作算子组成, 即变异算子、交叉算子和选择算子^[1]。

假设问题是 D 维的, 种群大小为 N , 对于第 G 代的每个个体 $\mathbf{X}_{i,G} = \{x_{1i,G}, x_{2i,G}, \dots, x_{Di,G}\}$, 对应的变异向量 $\mathbf{V}_{i,G} = \{v_{1i,G}, v_{2i,G}, \dots, v_{Di,G}\}$ 由如下的变异算子生成:

$$\mathbf{V}_{i,G} = \mathbf{X}_{r_1,G} + F \cdot (\mathbf{X}_{r_2,G} - \mathbf{X}_{r_3,G}) \quad (1)$$

式中 $r_1, r_2, r_3 \in [1, D]$, 是互不相同的随机整数, 且与 i 不相同; $F \in [0, 2]$, 是差分向量的收缩因子。

为了保持种群的多样性, 由目标向量和变异向量按照如下规则执行交叉操作产生试验向量 $\mathbf{U}_{i,G} = \{u_{1i,G}, u_{2i,G}, \dots, u_{Di,G}\}$:

$$u_{ji,G} = \begin{cases} v_{ji,G}, & \text{if } (\text{rand}_j[0,1] \leq CR) \text{ or } (j = j_{rand}) \\ x_{ji,G}, & \text{otherwise} \end{cases} \quad (2)$$

式中 $rand_j[0,1] \in [0,1]$ 为向量第 j 个分量对应的随机数; $CR \in [0,1]$ 为交叉概率; $j_{rand} \in [1,D]$ 为第 i 个向量对应的随机整数,用来确保试验向量 $U_{i,G}$ 至少有一个分量由对应的变异向量 $V_{i,G}$ 贡献。

DE 的选择算子采用贪婪选择策略,它保证新一代个体的适应值不比上一代的差。假设问题为最小化问题,操作如下:

$$x_{i,G+1} = \begin{cases} u_{i,G}, & \text{if}(f(U_{i,G}) < f(X_{i,G})) \\ x_{i,G}, & \text{otherwise} \end{cases} \quad (3)$$

2 改进的差分进化算法(MDE)

2.1 “位变”收缩因子

为了获得问题的全局最优解,既要求优化算法在设计空间内广泛地搜索,能找到全局最优解的大致范围,又要求算法能在此范围内高效地搜索,具有较高的寻优精度。基本的差分进化算法常会出现算法停滞和陷入局部最优的现象,其主要原因是 DE 采用固定的搜索因子和交叉概率。对于不同的优化对象,通常较难确定如何选择算法参数才能获得较好的结果。若采用较大的搜索因子和较小的交叉概率,DE 虽然能很好地保持种群的多样性,但其收敛速度和寻优精度会降低;反之如果采用较小的搜索因子和较大的交叉概率,其收敛速度较快,但容易陷入局部最优。文献[3]提出的“时变”收缩因子使收缩因子随迭代次数线性递减,在算法初期采用较大的收缩因子保持种群的多样性,随着迭代的进行,收缩因子逐渐减小,局部寻优能力逐渐加强。但是对于实际问题,常常难以确定合适的最大迭代次数;而且对于某些工程问题,往往只要求尽快地获得满意的改进解,因而在算法初期一味地采用较大的收缩因子是不合适的。为此,本文提出一种“位变”收缩因子。

“位变”收缩因子的基本思想是在迭代的每一步,不同的个体根据其适应值信息取不同的收缩因子,较好的个体取较小的收缩因子,使其在局部邻域内搜索,有更高的概率获得改进解,而较差的个体取较大的收缩因子,使种群保持多样性。具体的操作方法是,将个体按照适应值大小由高到低排序,各个体的收缩因子根据其排列位置线性变化。收缩因子的计算公式如下:

$$F_i = F_{\min} + \frac{i}{N}(F_{\max} - F_{\min}) \quad (4)$$

(4)式中 i 是对种群个体按照适应值从大到小排序

后个体的排列位置, N 为种群大小, F_{\max} 和 F_{\min} 分别为收缩因子的最大值和最小值,且满足 $0 \leq F_{\min} \leq F_{\max} \leq 2$ 。

2.2 正态分布的应用

DE 是一种自组织最小化的优化算法,随着迭代的进行,种群逐渐向最优个体聚集,个体间的差异减小,跳出局部最优的机会也减少。为了保持种群的多样性,本文采用正态分布函数对算法参数进行扰动,其基本思想是,对于种群中的每一个体,都采用正态分布函数扰动收缩因子 F 和交叉概率 CR ,使 F 和 CR 既有较高的概率在均值附近取值,也有一定的概率取得远离均值的值。这样算法既可以保持种群的多样性,也可以增强局部寻优能力。采用此方法的收缩因子的计算公式如下:

$$F_i = N(F_{ai}, \sigma) \quad (5)$$

式中 F_i 表示第 i 个个体的收缩因子, N 表示正态分布随机函数, F_{ai} 表示第 i 个个体的收缩因子的均值, σ 表示方差。交叉概率的扰动与(5)式相似。

2.3 DE/RBC/1/Bin 变异算子

基本的差分进化算法的变异算子采用一个基本向量与一个差分向量相加的形式。基本向量的选取会影响算法的性能,文献[5]研究了各种变异算子的特点,选择到当前代为止整个种群的最优点 X_{gbest} 作为基本向量有利于加快收敛,但容易陷入局部最优;选择自己 X_i 有利于在局部邻域内搜索;随机选取的基本向量有利于保持种群的多样性。文献[4]提出的三角变异算子能加快收敛速度,适用于局部寻优。为了使变异产生的变异向量既不过于分散,也不过分地向当前个体或最优个体聚集,本文综合了上述各种算子的特点,提出一种新的变异算子。

$$V_{i,G} = (a \cdot X_{r1,G} + b \cdot X_{gbest} + c \cdot X_{i,G}) / ((a + b + c) + F \cdot (X_{r2,G} - X_{r3,G})) \quad (6)$$

式中 $a, b, c = rand[0,1] \in [0,1]$, 即 a, b 和 c 均为 $[0,1]$ 内的随机数;其他参数的含义与(1)式相同。该算子的基本向量位于由 X_{gbest}, X_i 和 $X_{r1,G}$ 围成的三角形区域内,其具体位置由 a, b 和 c 三个随机数确定。根据差分变异算子的命名规则,(4)式为 DE/Rand/1/Bin;由于(9)式中基本向量由 X_{gbest}, X_i 和 $X_{r1,G}$ 构成,因此命名为 DE/RBC/1/Bin,其中 RBC 分别代表 Rand, Best 和 Current。

本文的 MDE 采用(5)式计算收缩因子的均值,并用(4)式扰动;交叉概率也用(5)式计算,但其均值

由“时变”方法^[6]计算;(6)式作用于种群中随机选取的部分较优个体,其它的个体由(1)式进行差分变异,其规则如下:

$$\begin{cases} \text{DE/RBC}/1/\text{Bin} & \text{rand}[0,1] \geq i/N \& \text{rand}[0,1] < 0.5 \\ \text{DE/Rand}/1/\text{Bin} & \text{else} \end{cases} \quad (7)$$

3 算法测试

为了验证 MDE 的性能,采用 Rosenbrock, Rastrigin 和 Griewank 等 3 个 Benchmark 测试函数^[7]分别在 10 维、20 维和 30 维的情况下对 MDE 测试,并与基本的 DE 进行比较。Rosenbrock 函数是单峰函数,但

其全局最优点隐藏于一条狭长的通道中不易获得,可考察算法的收敛速度和全局寻优能力; Griewank 和 Rastrigin 都是多峰函数,具有大量局部极值,可检验算法的全局搜索性能。

对于基本的 DE, $F = 0.5, CR = 0.4$; 本文的 MDE 的参数设置如下: $F \in [0.3, 0.7], \sigma_F = 0.2, CR \in [0.1, 0.4], \sigma_{CR} = 0.2$; 种群规模均取为 $N = 40$ 。

表 1 列出了两种算法对 3 个测试函数分别运行 50 次后得到的最优结果的平均值和方差。图 1~3 给出了在 30 维的情况下分别采用 DE 和 MDE 两种算法时,3 个测试函数的平均适应值随迭代次数的变化过程。

表 1 DE 和 MDE 分别运行 50 次的平均结果比较

Table 1 Comparison between DE and MDE on the three functions. The results are averaged over 50 runs

函数	维数	最大代数	DE	MDE
			均值(方差)	均值(方差)
R_n	10	1000	4.61E+00(1.19E+00)	2.58E+00(1.54E+00)
	20	2000	1.42E+01(1.11E+00)	1.44E+01(1.16E+01)
	30	3000	2.37E+01(7.32E-01)	2.32E+01(7.40E+00)
R_n	10	1000	0(0)	0(0)
	20	2000	2.67E+01(3.28E+00)	0(0)
	30	3000	8.09E+01(7.40E+00)	0(0)
G_r	10	400	7.13E-02(4.39E-02)	2.75E-02(2.12E-02)
	20	800	2.99E-10(1.06E-09)	0(0)
	30	1200	2.98E-12(9.78E-12)	0(0)

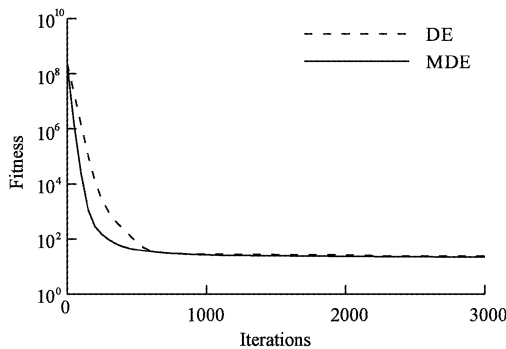


图 1 Rosenbrock 函数的收敛曲线

Fig.1 Comparison of convergence curves for Rosenbrock

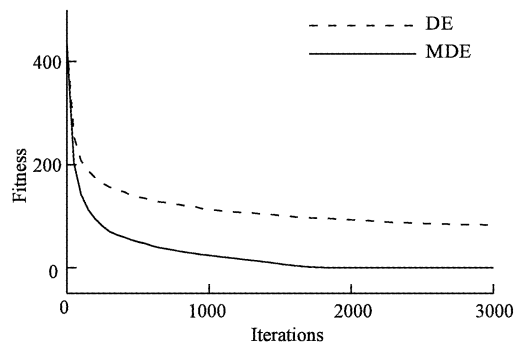


图 2 Rastrigin 函数的收敛曲线

Fig.2 Comparison of convergence curves for Rastrigin

由表 1 的结果可知,基本 DE 和 MDE 对 20 维和 30 维的 Rosenbrock 函数获得了基本相当的结果,而对其它所有情况,MDE 的结果比基本的 DE 更好。由图 1~3 的结果可以看出,MDE 与 DE 对 Rosenbrock 函数的收敛精度相当,但在算法初期 MDE 比 DE 的收敛速度更快,对于其他几个测试函数,MDE 比 DE 有更快的收敛速度和更高的收敛精度。

4 MDE 在涡轮气动优化中的应用

涡轮气动优化问题是复杂的非线性优化问题,设计变量多、高度非线性和计算成本高是该问题的主要特点。用于描述流场性质的 N-S 方程存在高度的非线性特征,各种激波、湍流、分离等复杂流动现象的综合使数值仿真呈高度的复杂性,涡轮几何模型的复杂性决定了数值计算的高成本。一方面为

验证 MDE 的性能,另一方面为提高某型涡轮的性能,本文将 MDE 用于涡轮的气动优化。

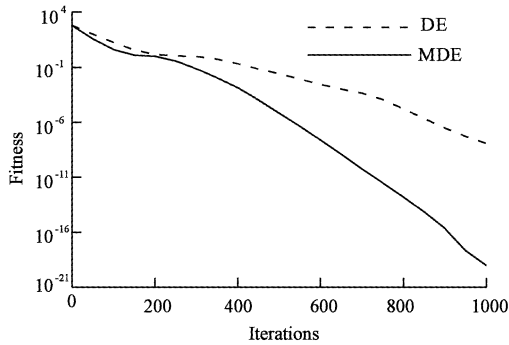


图 3 Griewank 函数的收敛曲线

Fig. 3 Comparison of convergence curves for Griewank

4.1 优化模型

本文的火箭发动机二级冲击式涡轮为直叶片,定子和转子分别为三列和两列。文中针对两级转子叶片进行优化,目标函数为涡轮效率,可由下式计算:

$$\eta = N_T / L_{ad} \quad (8)$$

式中 η 为涡轮效率, N_T 为涡轮有效功率, L_{ad} 为涡轮绝热功, N_T 和 L_{ad} 可分别由以下两式计算:

$$N_T = (M_1 \cdot n_1 + M_2 \cdot n_2) \cdot 2\pi N \quad (9)$$

$$L_{ad} = \frac{k}{k-1} RT_0 \left(1 - \frac{1}{\delta^{(k-1)/k}} \right) \dot{m} \quad (10)$$

上两式中 M 为燃气作用在动叶上的沿轴向的力矩,可由仿真计算得到, n 为叶片数目,下标 1 和 2 分别表示第一级动叶和第二级动叶, N 为转速。 k 为绝热指数, R 为燃气气体常数, T_0 为燃气入口总温, $\delta = p_0^* / p$ 为涡轮压比, \dot{m} 为涡轮流量。

叶片的参数化造型方法为:采用圆弧构造涡轮的前后缘, B 样条曲线^[8]构造叶背和叶盆。设计变量有:前后缘圆弧的半径和圆心角,前后缘构造角,叶型安装角、弦长以及样条曲线的控制点坐标等。

4.2 优化流程

全三维流场计算的网格数约 120 万,单机单次仿真用时约 15 小时。问题的困难在于设计变量多,仿真计算量大,计算成本高。为缩短计算时间,对优化流程并行处理,将 MDE 种群的各个设计点分配到不同的计算节点进行并行仿真,然后收集各计算节点的结果并根据 MDE 的算法规则产生下一代设计点。优化流程如图 4 所示。

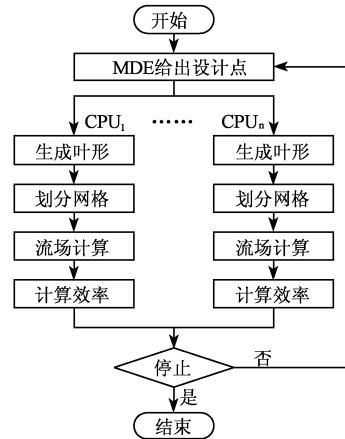


图 4 优化流程

Fig. 4 Optimization flow

4.3 优化结果及分析

经 30 粒 CPU 约 30 天的并行优化,得到了如图 5 所示的叶形。优化前后涡轮中径面上的压强分布如图 6 所示。

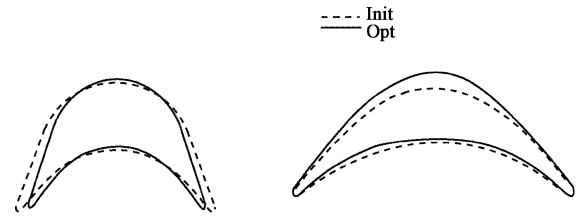


图 5 优化前后的叶形

Fig. 5 Comparisons of the shapes between the initial design and final optimized design

由图 6(a)可知,优化后第一级动叶栅绕流的平均压强略有增大,压强分布更光滑,原有的流通截面中部的附面层分离现象得到一定程度的改善。由图 6(b)可知,优化后第二级动叶栅绕流的平均压强略有减小,原有的处于叶片前部的附面层分离现象向叶片中部移动。计算表明,涡轮的总效率提高了 2.5%。

5 结论

本文对差分进化算法进行了改进:采用“位变”收缩因子加快获得改进解的速度;采用正态分布随机函数对收缩因子和交叉概率进行随机扰动来提高种群的多样性;提出一种新的变异算子提高算法的搜索精度。通过测试和应用,得到如下结论:

(1) Benchmark 函数的测试结果表明:该算法不仅在迭代初期比 DE 有更快的收敛速度,在算法后期也有较高的收敛精度,具有良好的综合性能。

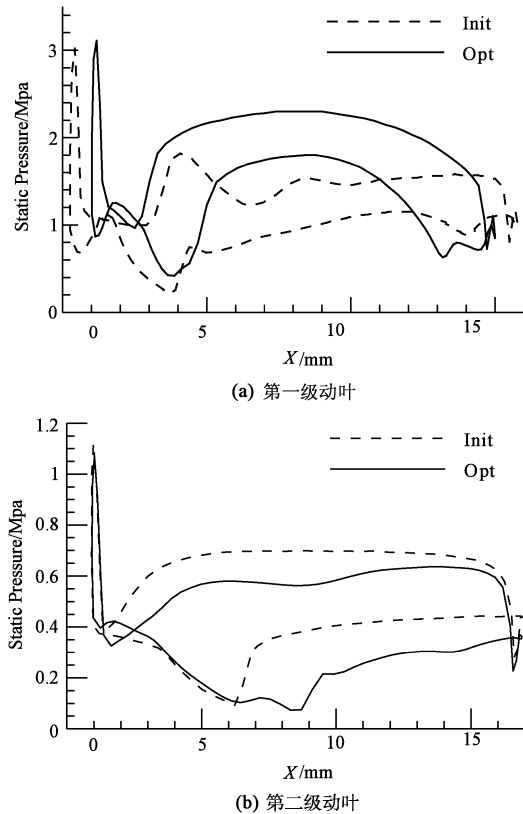


图6 优化前后叶片中径面上的压强分布

Fig.6 Comparison of pressure distribution on middle section for initial design and final optimized design

(2) 通过较小的计算成本,得到了某型火箭发动机涡轮叶片的改进叶形,使涡轮气动效率提高了2.5%。

(3) 该算法具有较快的收敛速度,能以较少的迭代次数获得优化问题较好的改进解,对于长耗时的工程仿真优化问题,能有效地节约计算成本。应用结果表明该算法是实用的。

参考文献:

- [1] Storn R, Price K V. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces[J]. *Journal of Global Optimization*, 1997, 11: 341 – 359.
- [2] Nasimul Noman, Hitoshi Iba, Accelerating differential evolution using an adaptive local search[J]. *IEEE Transactions on Evolutionary Computation*, 2008(2): 107 – 125.
- [3] Swagatam Das, Amit Konar, Uday K. Chakraborty, Two improved differential evolution schemes for faster global search[C]. *GECCO'05*, Washington, 2005: 991 – 998.
- [4] Fan H Y, Lampinen J. A trigonometric mutation operation to differential evolution[J]. *Journal of Global Optimization*, 2003, 27(1): 105 – 129.
- [5] Feoktistov V, Janaqi S. Generalization of the Strategies in differential evolution[C]. *Proceedings of 18th International Parallel and Distributed Processing Symposium*, 2004(7): 165 – 170.
- [6] 吴亮红,王耀南,袁小芳,等. 自适应二次变异差分进化算法[J]. *控制与决策*, 2006, 21(8): 898 – 902. [WU Liang-hong, WANG Yao-nan, YUAN Xiao-fang. Differential evolution algorithm with adaptive second mutation[J]. *Control and Decision*, 2006, 21(8): 898 – 902.]
- [7] Yao X, Liu Y, Lin G. Evolutionary programming made faster[J]. *IEEE Transactions on Evolutionary Computation*, 1999, 3(2): 82 – 102.
- [8] 施法中. 计算机辅助几何设计与非均匀有理B样条[M]. 北京:北京航空航天大学出版社,1994:228 – 270. [SHI Fa-zhong. *CAGD and NURBS*[M]. Beijing: BUAA Press, 1994: 228 – 270.]

作者简介:饶大林(1983-),男,博士研究生,研究方向为多学科设计优化及其在航天推进理论与工程中的应用。

通信地址:北京市海淀区学院路37号北京航空航天大学宇航学院宇航推进系新主楼B925(100191)

电话:(010)82314338

E-mail:cover0270@163.com

Modified Differential Evolutionary Algorithm for Fast Simulation Optimization and Its Application

RAO Da-lin, CAI Guo-biao

(School of Astronautics, Beijing University of Aeronautics and Astronautics, Beijing 100191, China)

Abstract: A modified differential evolutionary algorithm(MDE) is proposed. MDE adopts “Position” varying scale factor, which calculates the scale factor linearly according to the position of each individual after arranging by the fitness. To maintain good diversity, normal distribution function is used to disturb the parameters of MDE. A new mutation operator is proposed too, which can enhance the exploration efficiency and precision associating with basic mutation operator. The benchmark function result shows that the algorithm not only has good performance of exploration precision, but also has faster convergence speed than basic DE. At last, MDE is applied in aerodynamic optimization of turbine in LRE, and the result shows aerodynamic efficiency is increased by 2.5% with low computational cost. The application instance indicates good applicability of MDE for simulation optimization problem.

Key words: Differential evolution; “Position” varying scale factor; Normal distribution; Mutation operator; Simulation optimization; Turbine