

## 差分演化的收敛性分析与算法改进<sup>\*</sup>

贺毅朝<sup>1+</sup>, 王熙照<sup>2</sup>, 刘坤起<sup>1,3</sup>, 王彦祺<sup>1</sup>

<sup>1</sup>(石家庄经济学院 信息工程学院, 河北 石家庄 050031)

<sup>2</sup>(河北大学 数学与计算机学院, 河北 保定 071002)

<sup>3</sup>(中国地质大学 计算机学院, 湖北 武汉 430074)

### Convergent Analysis and Algorithmic Improvement of Differential Evolution

HE Yi-Chao<sup>1+</sup>, WANG Xi-Zhao<sup>2</sup>, LIU Kun-Qi<sup>1,3</sup>, WANG Yan-Qi<sup>1</sup>

<sup>1</sup>(School of Information Engineering, Shijiazhuang University of Economics, Shijiazhuang 050031, China)

<sup>2</sup>(College of Mathematics and Computer Science, Hebei University, Baoding 071002, China)

<sup>3</sup>(School of Computer, China University of Geosciences, Wuhan 430074, China)

+ Corresponding author: E-mail: heyichao@sjzue.edu.cn

He YC, Wang XZ, Liu KQ, Wang YQ. Convergent analysis and algorithmic improvement of differential evolution. *Journal of Software*, 2010,21(5):875-885. <http://www.jos.org.cn/1000-9825/3486.htm>

**Abstract:** To analyze the convergence of differential evolution (DE) and enhance its capability and stability, this paper first defines a differential operator (DO) as a random mapping from the solution space to the Cartesian product of solution space, and proves the asymptotic convergence of DE based on the random contraction mapping theorem in random functional analysis theory. Then, inspired by “quasi-physical personification algorithm”, this paper proposes an improved differential evolution with multi-strategy cooperating evolution (MEDE) is addressed based on the fact that each evolution strategy of DE has common peculiarity but different characteristics. Its asymptotic convergence is given with the definition of multi-strategy differential operator (MDO), and the connotative peculiarity of MEDE is analyzed. Compared with the original DE, DEfirDE and DEfirSPX, the simulation results on 5 classical benchmark functions show that MEDE has obvious advantages in the convergence rate, solution-quality and adaptability. It is suitable for solving complex high-dimension numeral optimization problems.

**Key words:** differential evolution; asymptotic convergence; contraction mapping; random operator; evolution strategy

**摘要:** 为了分析差分演化(differential evolution, 简称 DE)的收敛性并改善其算法性能, 首先将差分算子(differential operator, 简称 DO)定义为解空间到解空间的笛卡尔积的一种随机映射, 利用随机泛函理论中的随机压缩映射原理证明了 DE 的渐近收敛性; 然后, 在“拟物拟人算法”的启发下, 通过对 DE 各进化模式的共性特征与性能差异的分析, 提出了一种具有多进化模式协作的差分演化算法(differential evolution with multi-strategy cooperating

\* Supported by the National Natural Science Foundation of China under Grant Nos.60473045, 60471022 (国家自然科学基金); the Hebei Provincial Natural Science Foundation of China under Grant No.F2008000635 (河北省自然科学基金)

Received 2008-05-01; Accepted 2008-10-07; Published online 2009-09-11

evolution,简称 MEDE),分析了它所具有的隐含特性,并在多模式差分算子(multi-strategy differential operator,简称 MDO)定义的基础上证明了它的渐近收敛性.对 5 个经典测试函数的仿真计算结果表明,与原始的 DE,DEfirDE 和 DEfirSPX 等算法相比,MEDE 算法在求解质量、适应性和鲁棒性方面均具有较明显的优势,非常适于求解复杂高维函数的数值最优化问题.

关键词: 差分演化;渐近收敛性;压缩映射;随机算子;进化模式

中图法分类号: TP18 文献标识码: A

差分演化(differential evolution,简称 DE)<sup>[1]</sup>是由 Storn 和 Price 于 1996 年为求解切比雪夫多项式而提出的一种演化算法,在第 1 届 IEEE 演化大赛中表现超群,引起了国内外学者的广泛关注.目前,有关 DE 的研究已取得了许多进展,例如,文献[2]提出了一种自适应 Pareto 差分演化算法;文献[3]比较了 DE 与粒子群优化算法(particle swarm optimization,简称 PSO)和演化算法(evolution algorithm,简称 EA)的优劣;文献[4,5]基于不同策略分别提出了应用 DE 求解多目标优化问题的有效方法;文献[6]提出了一种求解组合优化问题的二进制编码差分演化算法;文献[7]将 DE 与局部搜索和单点交叉相结合,提出了求解高维函数优化问题的有效改进差分演化算法;文献[8]提出了一种基于混合优化策略的改进差分演化算法;文献[9]利用对群体个体的动态更新,提出了一种动态差分演化算法;文献[10]基于逆向优化(opposition-based optimization,简称 OBO)提出了一种逆向差分演化算法;文献[11]将三角变异嵌入到差分演化,较好地保持了 DE 收敛性与鲁棒性的平衡;文献[12]利用自适应局部搜索改善差分演化算法的收敛性能;文献[13]在 DE 中引入群体迁移操作,将其应用于电力系统中的无功优化.虽然 DE 的研究成果越来越多,但是关于其收敛性和收敛速率方面的理论研究还很少.为此,本文基于随机泛函理论,首先证明了 DE 的渐近收敛性,并在“拟物拟人算法”<sup>[14]</sup>思想的启发下,提出了一种简捷、高效的改进差分演化算法.

本文第 1 节简介 DE 原理及其特性.第 2 节利用随机压缩映射原理证明 DE 的渐近收敛性.第 3 节借鉴“拟物拟人算法”思想提出一种具有多进化模式协作的差分演化算法(differential evolution with multi-strategy cooperating evolution,简称 MEDE).随后,通过仿真实验对 MEDE 与 DE1,DE3,DE5,DEfirDE 和 DEfirSPX 等进行比较,指出 MEDE 在求解质量、适应性和鲁棒性方面具有更显著的优势.最后总结全文并提出今后的研究思路.

## 1 差分演化及其特性

差分演化(DE)<sup>[1]</sup>是一种基于实数编码的演化算法,在每一代演化迭代过程中,首先利用差分算子(differential operator,简称 DO)得到一个由父子混合个体构成的中间群体,然后由选择算子(selection operator,简称 SO)根据优胜劣汰的竞争机制重组新一代群体.当前,DE 有多种不同的进化模式(evolution strategy)<sup>[1,7]</sup>,其差别主要体现在差分算子的变异公式的不同.下面仅以 DE/t/1/bin 进化模式为基础,基于文献[6]中 DO 和 SO 的定义简述 DE 的算法流程与特性.

本文总假定  $f(X)$  为最小优化函数,解空间  $S = \{X | X = (x_1, x_2, \dots, x_d) \wedge L_j \leq x_j \leq U_j, j = 1, 2, \dots, d\}$  为  $d$  维欧氏空间  $R^d$  的有界子空间.令 DE 的第  $t$  代群体为  $P(t)$ ,其规模为  $s$ ,其中第  $i$  个基本个体用  $X_i(t) = (x_{i1}(t), x_{i2}(t), \dots, x_{id}(t)) \in S$  表示;相应于  $P(t)$  的中间群体由  $Q(t)$  表示,其中第  $i$  个中间个体为  $d$  维向量  $V_i(t) = (v_{i1}(t), v_{i2}(t), \dots, v_{id}(t)) \in S (1 \leq i \leq s)$ .设  $X_{best}(t)$  为群体  $P(t)$  中的最优个体,MAXT 为算法的最大迭代次数,则 DE 的算法描述如下:

算法 1. DE Algorithm<sup>[6]</sup>.

1. 随机生成初始群体  $P(0) = \{X_i(0) | 1 \leq i \leq s\}$ ,并置  $T=0$ ;
2. While ( $T < \text{MAXT}$ ) Do
3. For  $i=1$  to  $s$  Do
4. 随机选择  $p_1, p_2, p_3 \in \{1, 2, \dots, s\} - \{i\}$ ;
5. For  $j=1$  to  $d$  Do
6. 随机生成  $r \in (0, 1)$  与  $R(i) \in \{1, 2, \dots, d\}$ ;
7. 如果  $r \leq CR \vee j = R(i)$ , 则  $v_{ij}(T+1) = x_{p_1, j}(T) + \alpha(x_{p_2, j}(T) - x_{p_3, j}(T))$ , 否则  $v_{ij}(T+1) = x_{ij}(T)$ ;

8. Next  $j$
9. 如果  $f(V_i(T+1)) < f(X_i(T))$ , 则  $X_i(T+1) = V_i(T+1)$ , 否则  $X_i(T+1) = X_i(T)$ ;
10. Next  $i$
11. 计算  $P(T+1)$  中的最优个体  $X_{best}(T+1)$ , 并置  $T = T+1$ ;
12. EndWhile
13. 输出  $X_{best}(MAXT)$  和  $f(X_{best}(MAXT))$ , 算法结束.

由算法 1 可知, DE 迭代一次的时间复杂度为  $O(N^2)$ , 其中  $N = \max\{s, d\}$ . 令  $\Delta p_1 p_2 = X_{p_1} - X_{p_2}$  ( $p_1 \neq p_2$ ), 称  $\Delta p_1 p_2$  是由  $X_{p_1}$  和  $X_{p_2}$  构成的差异向量. 如果  $X_{p_1}$  和  $X_{p_2}$  是群体中随机选取的, 则称  $\Delta p_1 p_2$  为随机差异向量. 在 DE 的 DE/r/1/bin 模式中, 若将个体  $X_{p_1}$  称为基准个体, 则 DE/r/1/bin 模式的 DO 变异公式为基准个体  $X_{p_1}$  与随机差异向量  $\Delta p_1 p_2$  的线性组合. 事实上, DE 各模式均由基准个体与(随机)差异向量的线性组合构成.

DE 具有演化算法的两个基本特性: 隐含并行性和全局信息的有效利用<sup>[15]</sup>, DE 的 DO 还具有变异特性<sup>[1]</sup>. 此外, DE 具有学习与继承特性: 中间个体  $V_i(t+1)$  中平均约  $CR \times d + 1$  个分量来自  $X_{p_1}(t) + \alpha \times (X_{p_2}(t) - X_{p_3}(t))$ , 表明  $V_i(t+1)$  通过向个体  $X_{p_1}(t)$ ,  $X_{p_2}(t)$  和  $X_{p_3}(t)$  学习而产生; 又因  $1 - (1 - CR - 1/d)^d < 1$ , SO 产生新个体  $X_i(t+1)$  时必然至少有一维分量从  $X_i(t)$  中继承, 因此, DE 的父子群体间保持了部分信息的继承性. DE 具有自由探索性: 在 DO 中, 由于随机差异向量  $\Delta p_1 p_2$  是无约束随机选取的, 它必将导致由 DO 和 SO 产生的新个体也具有无约束的随机性, 使得 DE 能够以相对自由的方式生成新一代群体. DE 的自由探索性使其搜索范围更广, 有利于提高算法逃离局部极值陷阱的能力. 通过对 DE 特性的分析, 容易得出下面的结论:

**结论 1.** DE 的差分算子 DO 实质上是一种变异操作, 即个体  $X_i$  至少以  $1 - (1 - CR - 1/d)^d$  概率变异生成中间个体  $V_i$ ; DE 的选择算子 SO 是一种严格基于优胜劣汰策略的算子, 它通过淘汰  $X_i$  和  $V_i$  中较差个体产生更优的新一代个体. 对于最小优化问题, DE 评价函数序列  $\{f(X_i(t))\}_{1 \leq t \leq MAXT}$  为单调非递增序列.

## 2 差分演化的收敛性分析

在演化算法领域中, 有关遗传算法(genetic algorithm, 简称 GA)收敛性的研究成果颇丰, 目前已经提出了多种方法, 如 Vose-Liepins 模型法、公理化方法、基于 Markov 链方法和基于随机泛函分析的方法<sup>[16]</sup>. 在深入研究随机泛函分析方法时我们发现, 此方法更适合于证明 DE 的渐近收敛性, 其原因如下:

在利用随机泛函分析方法证明 GA 收敛性时, 由于 GA 的选择操作不能保证子代群体一定优于其父代群体, 因此必须采用精英保留策略(elitist strategy, 简称 ES)<sup>[16]</sup> 才能使所产生的子代群体总体上优于父代群体, 从而保证评价函数序列是单调非递增的. 由结论 1 可知: DE 的 SO 是严格基于优胜劣汰策略的, 它每次迭代产生的子代群体总是优于父代群体和中间群体, 从而评价函数序列一定是单调非递增的, 所以 DE 不需要 ES; 此外, DE 的 DO 是一种变异操作, 它通过基准个体与差异向量的线性组合进行变异, 其变异实质上与 GA 是相同的, 都是为了产生更优的个体以保证算法的不断进化. 正是基于上述原因, 以下利用随机泛函理论证明 DE 的渐近收敛性.

在计算机中进行数值计算时总要受到计算精度的限制, 为此, 不妨设 DE 的计算精度保留到小数点后第  $k$  位数字. 不失一般性, 令  $\min\{f(X) | \forall X \in S, 0 < f(X) < +\infty \text{ 且 } f(X) \text{ 不为常数}\}$  为最小优化问题,  $P$  为 DE 的群体, 其个体编码是有  $k$  位小数的  $d$  维实向量, 则解空间  $S = \prod_{i=1}^d [L_i, U_i]$  是由每一维分量都有  $k$  位小数的  $d$  维实向量全体构成的集合, 其中  $L_i < U_i$  且为实数. 令  $U = \max\{U = \max\{\overline{U}_i | \overline{U}_i = \lceil U_i \rceil \wedge 1 \leq i \leq d\}, L = \min\{\overline{L}_i | \overline{L}_i = \lfloor L_i \rfloor \wedge 1 \leq i \leq d\}$ , 则

$$|S| \leq \prod_{i=1}^d [(U_i - L_i)10^k + 1] \leq [10^k(U - L) + 1]^d.$$

在每次迭代中, 由于 DE 依次利用 DO 和 OS 实现进化, 故利用随机泛函理论可以将一次迭代抽象为由 DO 和 OS 合成的随机映射.

**定义 1.** 差分算子 DO 是一种按照概率  $CR + 1/d$  对个体向量的每一维分量进行重组变换的过程. 它是解空间上的一种随机映射  $\Psi_1: \Omega \times S \rightarrow S^2$ , 定义为

$$\mu\{\omega | \Psi_1(\omega, X) = \langle X, V \rangle\} = \mu\{V = X_i + \alpha(X_j - X_k)\} = 1 - (1 - CR - 1/d)^d,$$

其中,  $(\Omega, \mathcal{A}, \mu)$  为完全概率测度空间,  $\Omega$  为非空抽象集合, 其元素  $\omega$  为基本事件;  $\mathcal{A}$  是  $\Omega$  的某些子集所构成的  $\sigma$ -代数,  $\mu$  是  $\mathcal{A}$  上的概率测度;  $X_i, X_j, X_k$  为不同于  $X$  的个体,  $CR \in (0, 1)$  为变异因子,  $d$  为个体向量的维数.

**定义 2.** 选择算子 SO 是一种按照优胜劣汰的贪心策略从两个个体中选取最优个体的过程. 它是解空间上的映射  $\Psi_2: S^2 \rightarrow S$ . 对任意两个个体  $X$  与  $Y$ , 在  $\Psi_2$  作用下有

$$\Psi_2((X, Y)) = Z \wedge Z \in \{X, Y\} \wedge f(Z) = \min\{f(X), f(Y)\}.$$

在 DE 的一次迭代中, 相当于将映射  $\Psi = (\Psi_2 \circ \Psi_1): \Omega \times S \rightarrow S$  作用于当前群体  $P$ , 其中,  $\Psi$  是 DO 对应的随机映射和 SO 对应映射的逆序合成, 即  $P(t+1) = \Psi(\omega, P(t)) = \Psi_2(\Psi_1(\omega, P(t)))$ ,  $0 \leq t \leq \text{MAXT}-1$ . 设  $f(X_{\text{best}}(t))$  为  $P(t)$  中最优个体  $X_{\text{best}}(t)$  的适应值, 由于在  $\Psi$  的作用下, DE 每次迭代所产生的新一代群体必优于其父代群体, 因此, 各代群体中最优个体的适应值构成的序列  $\{f(X_{\text{best}}(t))\}_{1 \leq t \leq \text{MAXT}}$  必然是一个单调非递增序列. 因为算法的进化趋势可以利用最优个体刻画, 所以映射  $\Psi$  可重新定义为  $X_{t+1} = \Psi(\omega, X_t) = \Psi_2(\Psi_1(\omega, X_t))$ , 其中  $X_t$  和  $X_{t+1}$  分别为  $P(t)$  和  $P(t+1)$  中的最优个体,  $0 \leq t \leq \text{MAXT}-1$ .

**引理 1.** 设  $\lambda: S \times S \rightarrow R$  是定义在  $S$  上的距离, 并且满足  $\lambda(X_i, X_j) = |f(X_i) - f(X_j)|$ ,  $\forall X_i, X_j \in S$ , 则  $(S, \lambda)$  是一个完备可分的度量空间.

类似于文献[16]中的方法, 易证引理 1 成立.

**定理 1.** DE 的一次迭代所形成的随机映射  $\Psi$  是一个随机压缩算子.

证明: 根据 DO 算子与 SO 算子的定义以及结论 1, DE 每次迭代所产生的新群体比其父代群体总体上更优, 故对于随机映射  $\Psi = (\Psi_2 \circ \Psi_1): \Omega \times S \rightarrow S$ , 存在一个取值为非负实值的随机变量  $0 \leq K(\omega) < 1$ , a.s., 使得

$$\begin{aligned} \lambda(\Psi(\omega, X_{t-1}), \Psi(\omega, X_t)) &= \lambda(X_t, X_{t+1}) = |f(X_t) - f(X_{t+1})| \leq K(\omega) |f(X_{t-1}) - f(X_t)| = K(\omega) \lambda(X_{t-1}, X_t), \\ \Omega_0 &= \{\omega \mid \lambda(\Psi(\omega, X_{t-1}), \Psi(\omega, X_t)) \leq K(\omega) \lambda(X_{t-1}, X_t)\} \subseteq \Omega, \mu(\Omega_0) = 1. \end{aligned}$$

所以, DE 形成的映射  $\Psi: \Omega \times S \rightarrow S$  是一个随机压缩算子.  $\square$

**引理 2(随机压缩映射定理)**<sup>[16,17]</sup>. 设  $\Psi: \Omega \times S \rightarrow S$  为随机算子, 满足对几乎所有的  $\omega \in \Omega$ ,  $\Psi(\omega)$  均为压缩算子, 则  $\Psi(\omega)$  有唯一的随机不动点  $g(\omega)$ , 即  $\Psi(\omega, g(\omega)) = g(\omega)$ .

根据定理 1, DE 形成的随机映射  $\Psi$  是随机压缩算子, 再利用引理 2,  $\Psi(\omega)$  必存在唯一的随机不动点, 于是得到 DE 的收敛性定理如下:

**定理 2.** 设  $\Psi$  为 DE 形成的随机压缩算子, 则  $\Psi$  具有唯一的随机不动点, 即 DE 是渐进收敛的.

### 3 多进化模式差分演化算法及其实现

DE 存在多种进化模式(evolution strategy), 其一般表示形式为 DE/x/y/z<sup>[7]</sup>,  $x$  表示 DO 中与差异向量进行重组的基准个体是随机选取的还是当前群体的最优个体;  $y$  表示参与重组的差异向量个数;  $z$  表示重组所采用的方式, 主要有指数重组方式和二项式重组方式. 算法 1 中采用的是 DE/rand/1/bin 模式, 若将其中差分算子的变异公式作相应改变, 即可得到对应于不同 DO 模式的 DE 算法.

为便于分析, 下面给出基于二项式重组方式的 5 种进化模式的 DO 变异公式, 其中,  $X_{p_1} \sim X_{p_5}$  分别是父代群体中不同于  $X_i$  的 5 个个体,  $X_{\text{best}}$  是父代群体中的最好个体.

- (1) 在 DE/rand/1/bin 模式中, DO 的变异公式为  $V_i = X_{p_1} + \alpha \times (X_{p_2} - X_{p_3})$ ;
- (2) 在 DE/rand/2/bin 模式中, DO 的变异公式为  $V_i = X_{p_1} + \alpha \times (X_{p_2} - X_{p_3}) + \alpha \times (X_{p_4} - X_{p_5})$ ;
- (3) 在 DE/best/1/bin 模式中, DO 的变异公式为  $V_i = X_{\text{best}} + \alpha \times (X_{p_1} - X_{p_2})$ ;
- (4) 在 DE/best/2/bin 模式中, DO 的变异公式为  $V_i = X_{\text{best}} + \alpha \times (X_{p_1} - X_{p_2}) + \alpha \times (X_{p_3} - X_{p_4})$ ;
- (5) 在 DE/rand-to-best/1/bin 模式中, DO 的变异公式为  $V_i = X_i + \alpha \times (X_{\text{best}} - X_i) + \alpha \times (X_{p_1} - X_{p_2})$ .

在每一种进化模式中, 新个体  $V_i$  均由基准个体( $X_i, X_{p_1}$  或  $X_{\text{best}}$ )与差异向量重组而成. 这样, 根据基准个体和差异向量选取的不同, 我们将 5 种进化模式分成 3 类: 第 1 类模式为 DE/rand/1/bin 和 DE/rand/2/bin. 在此类中, 随机选择  $X_{p_1}$  为基准个体, 并由  $X_{p_1}$  与随机差异向量重组产生新个体  $V_i$ . 其特点是 DO 的自由探索性突出, 全局收敛性较强, 不易陷入局部最优, 但收敛速度较慢. 第 2 类模式为 DE/best/2/bin 和 DE/best/1/bin. 此类模式以当前

群体的最优个体  $X_{best}$  为基准个体,通过  $X_{best}$  与随机差异向量重组产生  $V_i$ ,其特点是 DO 的自由探索特性相对较弱,局部收敛性与继承性较强,收敛速度较快,但易陷入局部最优.第 3 类只有 DE/rand-to-best/1/bin 模式.它以  $X_i$  为基准个体,通过与固定差异向量( $X_{best}-X_i$ )和随机差异向量( $X_{p_1}-X_{p_2}$ )重组产生新个体  $V_i$ .特点是自由探索性和继承性相对保持均衡,具有较好的适应性,但鲁棒性相对较差.

进化模式除了性能上的差异之外还具有共性特征,即它们产生新个体的重组方式相同,均为基准个体与差异向量的线性组合,这使得各模式可以统一处理,便于模式间的协作进化.此外,各模式的变异公式中都至少含有 1 个随机差异向量,它使得当利用多种模式共同进化时,各模式之间可以通过随机差异向量交流信息.

**结论 2.** DE 的每一类进化模式既有各自的优势,也存在着不足,不能简单地说哪一类模式绝对地好,哪一类模式绝对地差.同时,各进化模式既存在结构和进化方式上的共性特征,又具有性能差异的特点,这种共性与差异使得它们可以协作进化.

黄文奇教授在文献[14]中提出的“拟物拟人算法”,其精髓在于牢牢地把握“智慧应源于大自然与自身的心灵”的思想,借鉴自然界中的各种现象以及人类生活中的各种社会经验,尤其是处理相互关系中的许多矛盾的各种攻关手腕都是很好的源泉<sup>[14]</sup>.在此启发下,借鉴政府部门为抑制物价非正常上涨往往同时采用临时价格干预、出台补贴政策、制定相关法规以及加强监管力度等多种措施和方法的社会经验,提出了一种基于多进化模式协作的差分演化算法(简称 MEDE).在 MEDE 中,同时使用 3 类模式中的一种进化模式,不妨设这 3 种进化模式分别为 DE/rand/1/bin,DE/best/1/bin 和 DE/rand-to-best/1/bin.MEDE 在每次迭代时,总是依次从 3 种模式中选取一种用于不同个体的进化,因此新一代群体必然是在 3 种模式共同作用下产生的.在 MEDE 中所涉及到的有关参数的含义与算法 1 相同,其算法流程描述如下:

**算法 2.** MEDE Algorithm.

1. 随机生成初始群体  $P(0)=\{X_i(0)|1\leq i\leq s\}$ ,并置  $T=0$ ;
2. While ( $T<MAXT$ ) Do
3. For  $i=1$  to  $s$  Do
4. 随机选择  $p_1,p_2,p_3\in\{1,2,\dots,s\}-\{i\}$ ;
5.  $M=i(\bmod 3)+1$ ,即选择进化模式类  $M\in\{1,2,3\}$ ;
6. For  $j=1$  to  $d$  Do
7. 随机生成  $r\in(0,1)$ 与  $R(i)\in\{1,2,\dots,d\}$ ;
8. 如果( $r>CR\wedge j\neq R(i)$ ),则  $v_{ij}(T+1)=x_{ij}(T)$ ,并转第 12 步继续;
9. 如果  $M=1$ ,则  $v_{ij}(T+1)=x_{p_1,j}(T)+\alpha(x_{p_2,j}(T)-x_{p_3,j}(T))$ ;
10. 如果  $M=2$ ,则  $v_{ij}(T+1)=x_{best,j}(T)+\alpha(x_{p_1,j}(T)-x_{p_2,j}(T))$ ;
11. 如果  $M=3$ ,则  $v_{ij}(T+1)=x_{ij}(T)+\alpha(x_{best,j}(T)-x_{ij}(T))+\alpha(x_{p_1,j}(T)-x_{p_2,j}(T))$ ;
12. Next  $j$
13. 如果  $f(V_i(T+1))<f(X_i(T))$ ,则  $X_i(T+1)=V_i(T+1)$ ,否则  $X_i(T+1)=X_i(T)$ ;
14. Next  $i$
15. 计算  $P(T+1)$ 中的最优个体  $X_{best}(T+1)$ ,并置  $T=T+1$ ;
16. EndWhile
17. 输出  $X_{best}(MAXT)$ 和  $f(X_{best}(MAXT))$ ,算法结束.

对于算法 2 中的第 8 步~第 11 步,在每次迭代时仅有 1 步被执行,并不增加额外的计算量,故 MEDE 迭代一次的时间复杂度也为  $O(N^2)$ , $N=\max\{s,d\}$ .显然,MEDE 仍然是在 DO 和 SO 的作用下通过迭代实现进化,只不过它同时使用 3 种不同的 DO 生成新一代群体而已.

为分析 MEDE 具有的隐含特性,在群体  $P$  上定义二元关系  $R$  如下:

$$R=\{\langle X_i,X_j\rangle|X_i\in P\wedge X_j\in P\wedge i\equiv j(\bmod 3)\}.$$

显然, $R$  是  $P$  上的等价关系, $P$  关于  $R$  的商集  $P/R=\{[X_1],[X_2],[X_3]\}$ .等价类  $[X_1]$  中的个体是利用 DE/rand/1/bin 模式进化的,等价类  $[X_2]$  中个体是利用 DE/rand/1/bin 模式进化的,而等价类  $[X_3]$  中的个体利用 DE/rand/1/bin 模式进化.这样, $[X_1],[X_2]$  和  $[X_3]$  相当于  $P$  的 3 个采用不同进化模式的子群体,因此,MEDE 具有隐含的多群体特性.

不妨设 $[X_1]=[X_2]=[X_3]=|P|/3$ ,注意到各进化模式的共性是其变异公式中至少含有 1 个随机差异向量,因此,等价类中个体在进化时,其 DO 变异公式的随机差异向量中至少有 1 个个体来自其他等价类的概率是 8/9,从而在产生的新个体中大约有 $(8/9)[1-(1-CR-1/d)^d]$ 的进化信息来自其他等价类中采用不同进化模式产生的个体.由此可见,随机差异向量在等价类之间起到了自然地交流与传递信息的作用,各等价类基于不同的模式进化,并以随机差异向量为载体进行信息交流与传递,使得它们的进化模式之间可以相互作用、相互促进.因此,MEDE 中所使用的 3 种进化模式以隐含的方式协作进化.

**结论 3.** MEDE 不但具有 DE 的所有特性,而且还具有多群体特性和多模式协作进化的特性;同时,MEDE 迭代一次的时间复杂度与 DE 相同,也为  $O(N^2)$ .

从算法 2 的描述可以看出,MEDE 是易于并行实现的.下面为了证明 MEDE 的渐近收敛性,引入多模式差分算子(multi-strategy differential operator,简称 MDO)的定义,并将其定义为解空间上的一种随机映射.

**定义 3.** 设  $V_i(t+1) \in Q(t+1)$ ,  $X_{p_1}(t), X_{p_2}(t), X_{p_3}(t) \in P(t)$ ,  $M \in \{1, 2, 3\}$ , 则多模式差分算子(MDO)的定义式为

$$v_{ij}(t+1) = \begin{cases} x_{p_1,j}(t) + \alpha \times (x_{p_2,j}(t) - x_{p_3,j}(t)), & \text{当}(r \leq CR \text{或} j = R(i)) \text{且} M = 1 \text{时} \\ x_{best,j}(t) + \alpha \times (x_{p_1,j}(t) - x_{p_2,j}(t)), & \text{当}(r \leq CR \text{或} j = R(i)) \text{且} M = 2 \text{时} \\ x_{ij}(t) + \alpha \times (x_{best,j}(t) - x_{ij}(t)) + \alpha \times (x_{p_1,j}(t) - x_{p_2,j}(t)), & \text{当}(r \leq CR \text{或} j = R(i)) \text{且} M = 3 \text{时} \\ x_{ij}(t), & \text{否则} \end{cases} \quad (1)$$

其中,  $p_1 \neq p_2 \neq p_3 \neq i$ ,  $x_{best,j}(t)$  为最优个体  $X_{best}(t)$  的第  $j$  维分量,参数  $\alpha, r, CR$  和  $R(i)$  的含义同算法 1.

**定义 4.** MDO 是一种按照概率  $CR+1/d$  对群体中个体向量的每一维分量进行重组变换的过程,它是解空间上的一种随机映射  $\Psi_3: \Omega \times S \rightarrow S^2$ , 可用下式定义:

$$\mu\{\omega | \Psi_3(\omega, X) = \langle X, V \rangle\} = \mu\{V = F(X, X_i, X_j, X_k, X_{best})\} = 1 - (1 - CR - 1/d)^d,$$

其中,  $F(X, X_i, X_j, X_k, X_{best})$  由公式(1)确定,  $CR \in (0, 1)$  为变异因子,  $d$  为个体向量的维数.

由于 MEDE 并没有改变 SO, 因此在每次迭代过程中相当于将映射  $\Psi' = (\Psi_2 \circ \Psi_3): S \rightarrow S$  作用于当前群体, 其中  $\Psi'$  是由 MDO 所对应的随机映射和 SO 所对应的映射逆序合成的, 即  $P(t+1) = \Psi'(P(t)) = \Psi_2(\Psi_3(\omega, P(t)))$ . 这样, 在映射  $\Psi'$  作用下, MEDE 各代群体中最优个体的适应值构成的序列  $\{f(X_{best}(t))\}_{1 \leq t \leq \text{MAXT}}$  是一个单调非递增序列. 同样地,  $\Psi'$  可重新定义为  $X_{t+1} = \Psi'(\omega, X_t) = \Psi_2(\Psi_3(\omega, X_t))$ ,  $X_t$  和  $X_{t+1}$  分别为  $P(t)$  和  $P(t+1)$  中的最优个体. 于是, 类似于定理 1, 可得如下定理:

**定理 3.** MEDE 的一次迭代所形成的随机映射  $\Psi'$  是一个随机压缩算子.

再根据引理 2, 即可得到 MEDE 渐近收敛的结论.

**定理 4.** 设  $\Psi'$  为 MEDE 形成的随机压缩算子, 则  $\Psi'$  具有唯一的随机不动点, 即 MEDE 是渐近收敛的.

上述所有分析和讨论也适用于指数重组方式. 此外, 为便于叙述, 以下将基于 DE/rand/1/bin, DE/best/1/bin 和 DE/rand-to-best/1/bin 模式的 DE 分别记为 DE1, DE3 和 DE5.

## 4 实验结果与分析

为了验证 MEDE 的有效性和鲁棒性, 对于表 1 中的 5 个典型高维 Benchmark 函数分别利用 MEDE 与 DE1, DE3, DE5 以及求解高维数值优化问题的有效算法 DEfirDE 和 DEfirSPX<sup>[7]</sup> 进行仿真计算, 从各种算法在 30 次独立运行中所求得的最佳解、最差解以及它们的数学期望和方差等多个方面进行比较, 并根据 De Jong 所提出的在线性能指标和离线性能指标<sup>[15]</sup>, 利用 MATLAB 语言绘制出部分仿真计算结果的平均在线性能曲线和平均离线性能曲线. 在所有仿真计算中, 使用微机的硬件环境为 DELL Pentium(R)4-CPU1.70 GHz, 128MB 内存, 并利用高级程序设计语言 VC++6.0 进行编程实现.

在对 MEDE, DE1, DE3 和 DE5 进行仿真计算时, 各算法的群体规模均设定为 50, 并取  $\alpha=0.5$ ,  $CR=0.3$ . 当函数  $f_1, f_2$  和  $f_4$  的维数  $n=30$  且  $f_3$  和  $f_5$  的维数  $n=10$  时, 算法的最大迭代次数设为 1 000; 当函数  $f_1, f_2$  和  $f_4$  的维数  $n=50$  且  $f_3$  和  $f_5$  的维数  $n=20$  时, 算法的最大迭代次数设为 2 000; 当函数  $f_1, f_2$  和  $f_4$  的维数  $n=100$  且  $f_3$  和  $f_5$  的维数  $n=30$  时, 算法的最大迭代次数设为 5 000. 表 2 和表 3 给出了对每个函数的各维情况独立计算 30 次所求得的最佳解

(best)、最差解(worst)及其数学期望(average)与标准差(standard deviation,简称为 Std.Dev.)的比较情况,其中计算误差为  $10^{-20}$ ,计算结果至少保留 3 位有效数字, $aEb$  表示  $a \times 10^b$ .为了进一步说明 MEDE 算法的有效性,在附录的图 1~图 10 中,分别给出了当  $f_1, f_2$  和  $f_4$  的维数  $n=100, f_3$  和  $f_5$  的维数  $n=30$  时,每种算法的平均在线性能曲线和平均离线性能曲线.

**Table 1** Five typical benchmark functions

表 1 5 个典型的 Benchmark 函数

| Problems            | $n$          | $[L_i, U_i]$        | Objective functions  | Optimal solution                 |
|---------------------|--------------|---------------------|--|----------------------------------|
| Sphere model        | 30, 50, 100  | $[-100, 100]$       | $f_1(X) = \sum_{i=1}^n x_i^2$  | $\min f_1(X)=0, X=(0,0,\dots,0)$ |
| Griewank function   | 330, 50, 100 | $[-600, 600]$       | $f_2(X) = \sum_{i=1}^n \frac{x_i^2}{4000} - \prod_{i=1}^n \cos \frac{x_i}{\sqrt{i}} + 1$   | $\min f_2(X)=0, X=(0,0,\dots,0)$ |
| Rastrigin function  | 10, 20, 30   | $[-5.12, 5.12]$     | $f_3(X) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$   | $\min f_3(X)=0, X=(0,0,\dots,0)$ |
| Ackley function     | 30, 50, 100  | $[-32.768, 32.768]$ | $f_4(X) = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}\right) - \exp\left[\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)\right] + 20 + \exp(1)$ | $\min f_4(X)=0, X=(0,0,\dots,0)$ |
| Rosenbrock function | 10, 20, 30   | $[-50, 50]$         | $f_5(X) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2]$   | $\min f_5(X)=0, X=(1,1,\dots,1)$ |

**Table 2** Computation results (1) of MEDE, DE1, DE3 and DE5

表 2 MEDE 与 DE1, DE3 和 DE5 的计算结果(1)

| Problems | $n$ | MEDE      |           |           |           | DE1       |           |           |           |
|----------|-----|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
|          |     | Best      | Worst     | Average   | Std.Dev.  | Best      | Worst     | Average   | Std.Dev.  |
| $f_1$    | 30  | 0.0       | 0.0       | 0.0       | 0.0       | 4.90E-12  | 2.54E-11  | 1.04E-11  | 6.55E-12  |
|          | 50  | 0.0       | 0.0       | 0.0       | 0.0       | 7.45E-14  | 1.38E-13  | 1.13E-13  | 2.39E-14  |
|          | 100 | 0.0       | 0.0       | 0.0       | 0.0       | 7.56E-13  | 2.17E-12  | 1.33E-12  | 4.67E-13  |
| $f_2$    | 30  | 0.0       | 0.0       | 0.0       | 0.0       | 8.47E-11  | 1.493E-9  | 4.62E-10  | 5.96E-10  |
|          | 50  | 0.0       | 0.0       | 0.0       | 0.0       | 7.34E-14  | 1.96E-13  | 1.27E-13  | 4.70E-14  |
|          | 100 | 0.0       | 1.11E-16  | 4.40E-19  | 5.30E-19  | 4.36E-13  | 8.73E-13  | 6.94E-13  | 1.65E-13  |
| $f_3$    | 10  | 0.0       | 0.0       | 0.0       | 0.0       | 0.0       | 0.0       | 0.0       | 0.0       |
|          | 20  | 0.0       | 0.0       | 0.0       | 0.0       | 2.836E-3  | 7.658 904 | 3.481 037 | 2.783 758 |
|          | 30  | 0.0       | 0.0       | 0.0       | 0.0       | 33.359 22 | 42.421 61 | 36.148 08 | 3.269 448 |
| $f_4$    | 30  | 1.33E-15  | 1.33E-15  | 1.33E-15  | 0.0       | 6.842E-7  | 1.375E-6  | 9.330E-7  | 2.333E-7  |
|          | 50  | 1.33E-15  | 1.33E-15  | 1.33E-15  | 0.0       | 7.009E-8  | 1.141E-7  | 8.470E-8  | 1.546E-8  |
|          | 100 | 1.33E-15  | 1.33E-15  | 1.33E-15  | 0.0       | 1.620E-7  | 2.001E-7  | 1.797E-7  | 1.402E-8  |
| $f_5$    | 10  | 1.052 914 | 4.235 053 | 3.334 665 | 1.064 147 | 1.427 750 | 7.338 329 | 4.378 123 | 2.200 322 |
|          | 20  | 12.254 27 | 13.484 14 | 12.987 83 | 0.445 826 | 13.121 24 | 15.566 62 | 14.119 63 | 0.730 292 |
|          | 30  | 19.059 66 | 20.216 97 | 19.777 94 | 0.452 870 | 21.087 23 | 30.583 40 | 24.425 69 | 3.563 306 |

**Table 3** Computation results (2) of MEDE, DE1, DE3 and DE5

表 3 MEDE 与 DE1, DE3 和 DE5 的计算结果(2)

| Problems | $n$ | DE3       |           |           |           | DE5       |           |           |           |
|----------|-----|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
|          |     | Best      | Worst     | Average   | Std.Dev.  | Best      | Worst     | Average   | Std.Dev.  |
| $f_1$    | 30  | 0.0       | 0.0       | 0.0       | 0.0       | 0.0       | 6.15E-17  | 1.23E-17  | 2.46E-17  |
|          | 50  | 0.0       | 0.0       | 0.0       | 0.0       | 3.939E-2  | 0.820 291 | 0.258 488 | 0.325 906 |
|          | 100 | 0.0       | 0.0       | 0.0       | 0.0       | 99.957 27 | 1 188.826 | 432.447 1 | 386.307 9 |
| $f_2$    | 30  | 9.865E-3  | 9.865E-3  | 9.865E-3  | 0.0       | 0.0       | 0.0       | 0.0       | 0.0       |
|          | 50  | 0.0       | 0.0       | 0.0       | 0.0       | 9.268E-2  | 9.268E-2  | 9.268E-2  | 0.0       |
|          | 100 | 1.04E-14  | 1.04E-14  | 1.04E-14  | 0.0       | 7.010 963 | 7.010 963 | 7.010 963 | 0.0       |
| $f_3$    | 10  | 0.0       | 1.989 918 | 0.497 480 | 0.759 913 | 0.0       | 0.0       | 0.0       | 0.0       |
|          | 20  | 0.994 959 | 10.944 54 | 5.140 621 | 3.007 819 | 1.55E-13  | 6.163E-8  | 0.198 992 | 0.397 984 |
|          | 30  | 6.964 713 | 12.934 46 | 9.1536 21 | 2.387 899 | 0.994 959 | 13.823 06 | 5.898 956 | 4.278 037 |
| $f_4$    | 30  | 1.33E-15  | 1.33E-15  | 1.33E-15  | 0.0       | 1.33E-15  | 3.251E-8  | 5.418E-9  | 1.211E-8  |
|          | 50  | 1.33E-15  | 1.155 149 | 0.731 599 | 0.539 401 | 7.299E-2  | 1.269 762 | 0.640 311 | 0.629 461 |
|          | 100 | 1.604 366 | 3.086 098 | 2.421 586 | 0.540 460 | 3.119 028 | 5.020 140 | 3.941 827 | 0.677 889 |
| $f_5$    | 10  | 1.667E-3  | 4.010 489 | 2.468 464 | 1.238 549 | 1.061 543 | 5.339 517 | 3.678 446 | 1.416 598 |
|          | 20  | 6.139 965 | 70.476 17 | 32.141 60 | 28.302 42 | 15.680 20 | 73.217 29 | 28.391 72 | 22.437 64 |
|          | 30  | 1.848E-4  | 27.610 35 | 19.734 78 | 7.889 145 | 22.442 14 | 146.563 8 | 61.203 06 | 47.743 48 |

从表 2 和表 3 的计算结果可以看出:对于函数  $f_1, f_2, f_3$  和  $f_4$ ,无论是在 30 次运算中所求得的最好解和最差解,还是它们的数学期望与标准差,MEDE 都远远优于 DE1,DE3 和 DE5,而且这种优势随着函数维数的增大越来越明显;对于函数  $f_5$ ,虽然 MEDE 求得的最好解比 DE3 稍微差些,但仍优于 DE1 和 DE5,而且 MEDE 求得的数学期望与标准差比 DE1,DE3 和 DE5 均优.在附录中,由图 1~图 10 给出的各算法平均在线性能曲线和平均离线性能曲线的比较也可以看出,MEDE 比 DE1,DE3 和 DE5 具有明显的优势.由此不难得出结论:MEDE 在保持算法时间复杂度不变的前提下,其求解质量、适应性和鲁棒性均比 DE1,DE3 和 DE5 更优,而且随着函数维数的逐渐增大,这种优势也越来越大.

在将 MEDE,DEfirDE 和 DEfirSPX 进行比较时,对于函数  $f_1, f_2$  和  $f_4$ ,设定  $\alpha=0.5$  且  $CR=0.3$ ,算法的最大迭代次数设为 50 000;对于函数  $f_3$  和  $f_5$ ,仍设定  $\alpha=0.5$  且  $CR=0.3$ ,但算法的最大迭代次数与文献[7]中相同.由于文献[7]中使用微机的硬件配置为 Intel Pentium Processor CPU1700 MHz,512 MB 内存,并在 Jbuilder X 环境下编程实现,因此,表 4 中关于 MEDE,DEfirDE 和 DEfirSPX 在解的数学期望与标准差方面的比较是有效的.

Table 4 Computation results of MEDE, DEfirDE and DEfirSPX

表 4 MEDE 与 DEfirDE,DEfirSPX 的计算结果

| Problems | $n$ | MEDE |           | DEfirDE   |     | DEfirSPX  |          |     |          |          |
|----------|-----|------|-----------|-----------|-----|-----------|----------|-----|----------|----------|
|          |     | $P$  | Average   | Std.Dev.  | $P$ | Average   | Std.Dev. | $P$ | Average  | Std.Dev. |
| $f_1$    | 100 | 50   | 1.0 E-20  | 0.0       | 100 | 1.0E-6    | 0.0      | 100 | 1.0E-6   | 0.0      |
|          | 200 | 100  | 1.0 E-20  | 0.0       | 200 | 17.678    | 9.483    | 200 | 0.856 8  | 0.256 3  |
| $f_2$    | 100 | 50   | 4.76E-16  | 5.49E-16  | 100 | 1.0E-6    | 0.0      | 100 | 1.0E-6   | 0.0      |
|          | 200 | 100  | 1.11E-16  | 0.0       | 200 | 0.598 4   | 0.141 9  | 200 | 0.163 1  | 0.031 4  |
| $f_3$    | 100 | 100  | 27.28548  | 3.405 744 | 100 | 1.0E-6    | 0.0      | 100 | 1.0E-6   | 0.0      |
|          | 200 | 200  | 111.089 1 | 1.643 677 | 200 | 0.145 3   | 0.277 1  | 200 | 0.002 4  | 0.001 1  |
| $f_4$    | 100 | 50   | 1.0 E-20  | 0.0       | 100 | 1.2E-6    | 6.07E-7  | 100 | 1.0E-6   | 0.0      |
|          | 200 | 100  | 1.33E-15  | 0.0       | 200 | 0.312 3   | 0.042 6  | 200 | 0.158 9  | 0.020 7  |
| $f_5$    | 100 | 100  | 9.230 016 | 11.027 59 | 100 | 107.560 4 | 28.252 9 | 100 | 99.108 6 | 18.573 5 |
|          | 200 | 200  | 109.227 4 | 68.868 32 | 200 | 5 302.79  | 2363.74  | 200 | 996.69   | 128.483  |

从表 4 中可以看出:对于函数  $f_3$ ,MEDE 的求解效果虽然差于 DEfirDE 和 DEfirSPX,但对于函数  $f_1, f_2$  和  $f_4$ ,在群体规模  $P$  仅为 DEfirDE 和 DEfirSPX 的 1/2、最大迭代次数仅为它们的 1/10 情况下,MEDE 所求得解的数学期望与标准差都明显优于 DEfirDE 和 DEfirSPX;而对于函数  $f_5$ ,在与 DEfirDE 和 DEfirSPX 具有相同群体规模和最大迭代次数的情况下,MEDE 所求得解的数学期望与标准差也显著优于 DEfirDE 和 DEfirSPX.因此,上述比较的结果表明:总体而言,MEDE 比 DEfirDE 和 DEfirSPX 的求解质量、适应性和鲁棒性更优,全局收敛性更好;此外,由于 MEDE 不需要增加额外的运算量,其求解速度也比 DEfirDE 和 DEfirSPX 更快.

## 5 结论与展望

本文首先基于随机压缩映射原理证明了 DE 的渐近收敛性,然后在“拟物拟人算法”思想的启发下,利用 DE 各个进化模式的特点,提出了一种具有多进化模式协作的差分演化算法,并分析了其复杂性和收敛性.最后,应用 5 个经典测试函数对 MEDE,DE1,DE3,DE5,DEfirDE 和 DEfirSPX 等进行仿真计算比较,根据比较结果得出,MEDE 在求解质量和适应性等方面具有更大的优势,非常适于求解连续域上复杂高维函数的数值优化问题.虽然本文在 DE 的渐近收敛性分析方面作了初步探讨,但有关 DE 收敛速率的理论分析还有待于进一步研究.此外,对于 MEDE 算法能否通过动态地分配各进化模式的使用比例或者与局部搜索策略相结合来进一步改进其收敛性能,也是今后需要进一步分析和探讨的问题.

致谢 衷心感谢审稿专家提供的宝贵修改意见和建议.

## References:

- [1] Storn R, Price K. Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces. Journal of Global Optimization, 1997,11(4):341–359.



- [2] Abbass HA. The self-adaptive Pareto differential evolution. In: Proc. of the 2002 Congress on Evolutionary Computation (CEC 2002), Vol.1. New York: IEEE Service Center, 2002. 831–836.
- [3] Vesterstrom J, Thomsen R. A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithm on numerical benchmark problems. In: Proc. of the IEEE 2004 Congress on Evolutionary Computation (CEC 2004, Vol.2. New York: IEEE Service Center, 2004. 1980–1987.
- [4] Babu BV, Jehan MML. Differential evolution for multiple-objective optimization. *Evolutionary Computation*, 2003,11(4):8–12.
- [5] Zhang LB, Zhou CG, Ma M, Sun CT. A multi-objective differential evolution algorithm based on max-min distance density. *Journal of Computer Research and Development*, 2007,44(1):177–184 (in Chinese with English abstract).
- [6] He YC, Wang XZ, Kou YZ. A binary differential evolution algorithm with hybrid encoding. *Journal of Computer Research and Development*, 2007,44(9):1476–1484 (in Chinese with English abstract).
- [7] Noman N, Iba H. Enhancing differential evolution performance with local search for high dimensional function optimization. In: Beyer HG, *et al.*, eds. Proc. of the 2005 Conf. on Genetic and Evolutionary Computation (GECCO 2005). New York: ACM Press, 2005. 967–974.
- [8] Zhao GQ, Peng XY, Sun N. A modified differential evolution algorithm with hybrid optimization strategy. *ACTA Electronica Sinica*, 2006,34(12):2402–2405 (in Chinese with English abstract).
- [9] Qing AY. Dynamic differential evolution strategy and applications in electromagnetic inverse scattering problem. *IEEE Trans. on Geoscience and Remote Sensing*, 2006,44(1):116–125.
- [10] Rahnamayan S. Opposition-Based differential evolution [Ph.D. Thesis]. Waterloo: University of Waterloo, 2007.
- [11] Fan HY, Lampinen J. A trigonometric mutation operation to differential evolution. *Journal of Global Optimization*, 2003,27(1): 105–129.
- [12] Noman N, Iba H. Accelerating differential evolution using an adaptive local search. *IEEE Trans. on Evolutionary Computation*, 2008,12(1):107–125.
- [13] Zhang FT, Song JH, Li J, Cheng XL. A hybrid differential evolution method for optimal reactive power optimization. *Power System Technology*, 2007,31(9):33–37 (in Chinese with English abstract).
- [14] Huang WQ, Xu RH. Introduction to the Theory of Recent Computation—Research on Background, Foreground and Algorithms of NP Hard Problems. Beijing: Science Press, 2004. 47–70 (in Chinese).
- [15] Liu Y, Kang LS, Chen YP. Nonnumerical Parallel Algorithms—Genetic Algorithm. Beijing: Science Press, 2003. 22–86 (in Chinese).
- [16] Li MQ, Kou JS, Lin D, Li SQ. The Foundational Theory and Application of Genetic Algorithms. Beijing: Science Press, 2003. 115–119 (in Chinese).
- [17] Lu TS. Random Functional Analysis and Its Application. Qingdao: Qingdao Ocean University Press, 1990. 44–100 (in Chinese).

#### 附中文参考文献:

- [5] 张利彪,周春光,马铭,孙彩堂.基于极大极小距离密度的多目标微分进化算法. *计算机研究与发展*,2007,44(1):177–184.
- [6] 贺毅朝,王熙照,寇应展.一种具有混合编码的二进制差分演化算法. *计算机研究与发展*,2007,44(9):1476–1484.
- [8] 赵光权,彭喜元,孙宁.基于混合优化策略的微分进化改进算法. *电子学报*,2006,34(12):2402–2405.
- [13] 张丰田,宋家骅,李鉴,程晓磊.基于混合差异进化优化算法的电力系统无功优化. *电网技术*,2007,31(9):33–37.
- [14] 黄文奇,许如初.近世计算理论导引——NP 难度问题的背景、前景及其求解算法研究.北京:科学出版社,2004.47–70.
- [15] 刘勇,康立山,陈毓屏.非数值并行算法——遗传算法.北京:科学出版社,2003.22–86.
- [16] 李敏强,寇纪松,林丹,李书全.遗传算法的基本理论与应用.北京:科学出版社,2003.115–119.
- [17] 卢同善.随机泛函分析及应用.青岛:青岛海洋大学出版社,1990.44–100.

#### 附录

当函数  $f_1, f_2$  和  $f_4$  的维数  $n=100$ ,  $f_3$  和  $f_5$  的维数  $n=30$  时,算法 MEDE, DE1, DE3 和 DE5 的平均在线性能曲线和平均离线性能曲线比较,如图 1~图 10 所示.

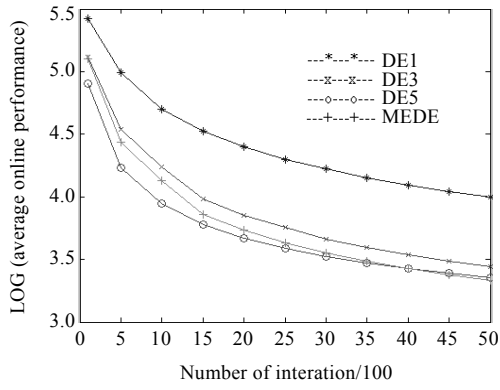


Fig.1 Comparison of  $f_1$ 's average online performance

图 1 求解  $f_1$  的平均在线性能比较

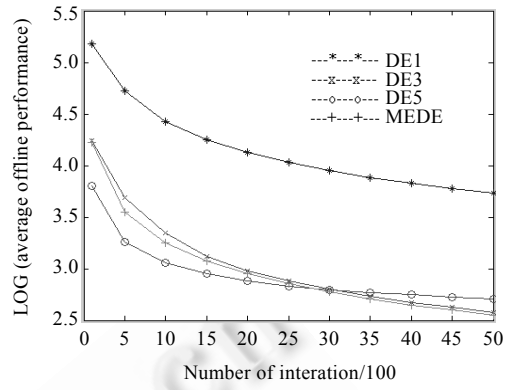


Fig.2 Comparison of  $f_1$ 's average offline performance

图 2 求解  $f_1$  的平均离线性能比较

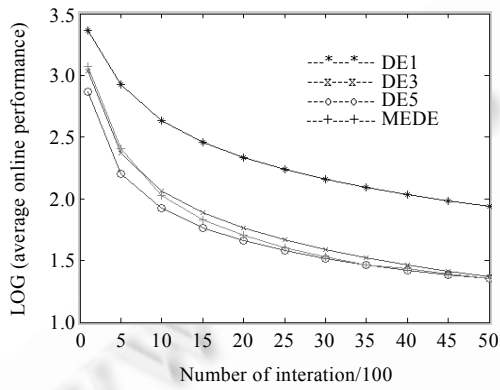


Fig.3 Comparison of  $f_2$ 's average online performance

图 3 求解  $f_2$  的平均在线性能比较

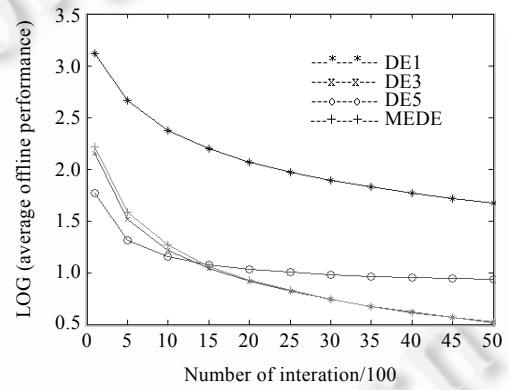


Fig.4 Comparison of  $f_2$ 's average offline performance

图 4 求解  $f_2$  的平均离线性能比较

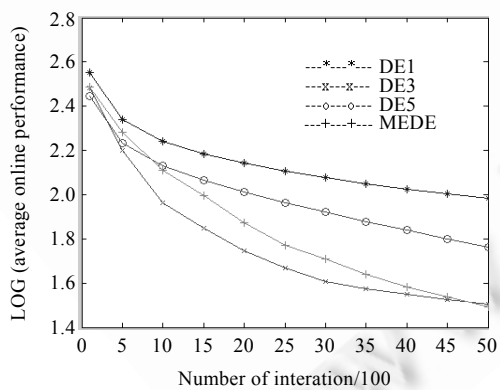


Fig.5 Comparison of  $f_3$ 's average online performance

图 5 求解  $f_3$  的平均在线性能比较

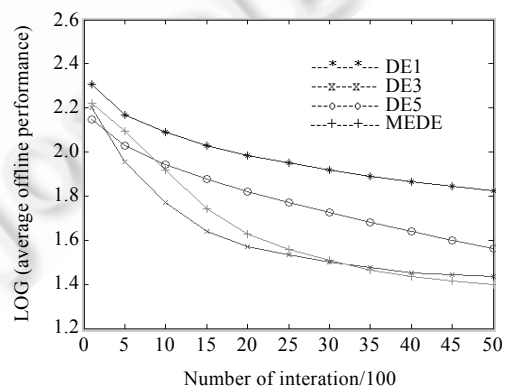


Fig.6 Comparison of  $f_3$ 's average offline performance

图 6 求解  $f_3$  的平均离线性能比较

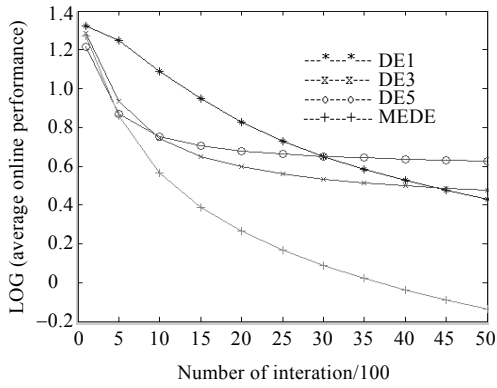


Fig.7 Comparison of  $f_4$ 's average online performance

图7 求解  $f_4$  的平均在线性能比较

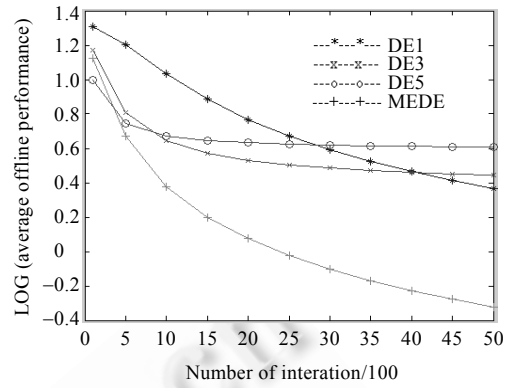


Fig.8 Comparison of  $f_4$ 's average offline performance

图8 求解  $f_4$  的平均离线性能比较

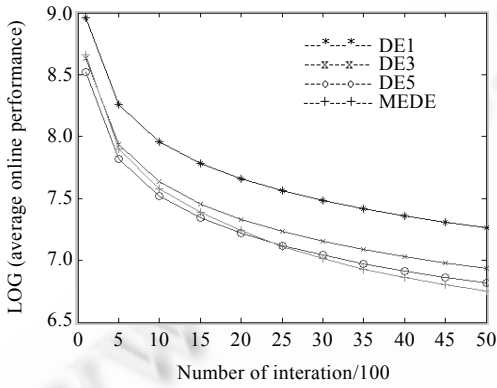


Fig.9 Comparison of  $f_5$ 's average online performance

图9 求解  $f_5$  的平均在线性能比较

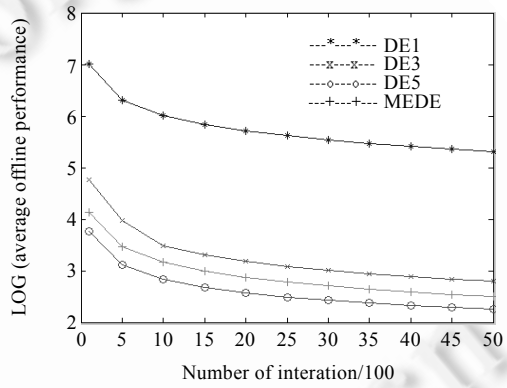


Fig.10 Comparison of  $f_5$ 's average offline performance

图10 求解  $f_5$  的平均离线性能比较



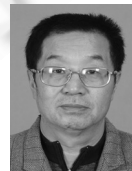
贺毅朝(1969—),男,河北晋州人,副教授,主要研究领域为智能计算,计算机密码学,计算复杂性理论.



刘坤起(1966—),男,博士生,副教授,主要研究领域为智能算法.



王熙照(1963—),男,博士,教授,博士生导师,主要研究领域为机器学习,计算智能.



王彦祺(1946—),男,教授,主要研究领域为智能计算,计算复杂性理论.