



随机交叉自学策略改进的教与学优化算法

黎延海, 雍龙泉, 拓守恒

引用本文:

黎延海, 雍龙泉, 拓守恒. 随机交叉自学策略改进的教与学优化算法[J]. 智能系统学报, 2021, 16(2): 313–322.

LI Yanhai, YONG Longquan, TUO Shouheng. Teaching-learning-based optimization algorithm based on random crossover-self-learning strategy[J]. *CAAI Transactions on Intelligent Systems*, 2021, 16(2): 313–322.

在线阅读 View online: <https://dx.doi.org/10.11992/tis.201910045>

您可能感兴趣的其他文章

布谷鸟搜索算法研究及其应用进展

Overview of the cuckoo search algorithm and its applications

智能系统学报. 2020, 15(3): 435–444 <https://dx.doi.org/10.11992/tis.201811005>

基于混沌搜索和权重学习的教与学优化算法及其应用

Teaching-learning-based optimization algorithm based on chaotic search and weighted learning and its application

智能系统学报. 2018, 13(5): 818–828 <https://dx.doi.org/10.11992/tis.201705017>

求解离散优化问题的元胞量子狼群演化算法

Cellular and quantum-behaved wolf pack evolutionary algorithm for solving discrete optimization problems

智能系统学报. 2018, 13(5): 716–727 <https://dx.doi.org/10.11992/tis.201705007>

一种求解多模态复杂问题的混合和声差分算法

Hybrid algorithm based on harmony search and differential evolution for solving multi-modal complex problems

智能系统学报. 2018, 13(2): 281–289 <https://dx.doi.org/10.11992/tis.201612030>

具有Levy变异和精英自适应竞争机制的蚁狮优化算法

Ant lion optimizer with levy variation and adaptive elite competition mechanism

智能系统学报. 2018, 13(2): 236–242 <https://dx.doi.org/10.11992/tis.201706091>

一种精英反向学习的萤火虫优化算法

Firefly optimization algorithm utilizing elite opposition-based learning

智能系统学报. 2017, 12(5): 710–716 <https://dx.doi.org/10.11992/tis.201706014>

微信公众平台



关注微信公众号, 获取更多资讯信息

DOI: 10.11992/tis.201910045

网络出版地址: <https://kns.cnki.net/kcms/detail/23.1538.TP.20200721.1319.004.html>

随机交叉-自学策略改进的教与学优化算法

黎延海¹, 雍龙泉^{1,2}, 拓守恒³

(1. 陕西理工大学 数学与计算机科学学院, 陕西 汉中 723001; 2. 陕西省工业自动化重点实验室, 陕西 汉中 723001; 3. 西安邮电大学 计算机学院, 陕西 西安 710121)

摘要: 针对非原点最优的复杂优化问题(最优解不在坐标原点), 提出了一种基于随机交叉-自学策略的教与学优化算法(teaching and learning optimization algorithm based on random crossover-self-study strategy, CSTLBO)。对标标准教与学优化算法的“教阶段”和“学阶段”的空间扰动进行了几何解释, 改进了原有的“教阶段”和“学阶段”, 并引入随机交叉策略和“自学”策略来提高算法的全局寻优能力。通过使用 20 个 Benchmark 函数进行仿真, 并与 6 种改进的教与学优化算法进行结果比较及 Wilcoxon 秩和检验分析, 结果表明 CSTLBO 算法能有效避免陷入局部最优, 具有良好的全局搜索能力, 求解精度高, 稳定性好。

关键词: 群体智能; 教与学优化; 随机交叉; “自学”策略; Benchmark 函数; 非原点最优; 多样性分析

中图分类号: TP391 **文献标志码:** A **文章编号:** 1673-4785(2021)02-0313-10

中文引用格式: 黎延海, 雍龙泉, 拓守恒. 随机交叉-自学策略改进的教与学优化算法[J]. 智能系统学报, 2021, 16(2): 313-322.

英文引用格式: LI Yanhai, YONG Longquan, TUO Shouheng. Teaching-learning-based optimization algorithm based on random crossover-self-learning strategy[J]. CAAI transactions on intelligent systems, 2021, 16(2): 313-322.

Teaching-learning-based optimization algorithm based on random crossover-self-learning strategy

LI Yanhai¹, YONG Longquan^{1,2}, TUO Shouheng³

(1. School of Mathematics and Computer Science, Shaanxi University of Technology, Hanzhong 723001, China; 2. Shaanxi Key Laboratory of Industrial Automation, Hanzhong 723001, China; 3. School of Computer Science and Technology, Xi'an University of Posts and Telecommunications, Xi'an 710121, China)

Abstract: This paper presents a teaching-learning-based optimization algorithm based on a random crossover-self-learning strategy (CSTLBO) for solving non-origin-optimal complex optimization problems in circumstances in which the optimum solution is outside the origin of the coordinates. First, the spatial perturbation in the “teaching” and “learning” stages of the standard TLBO algorithm is interpreted geometrically, and the original “teaching” and “learning” stages are improved. Then, random crossover and “self-learning” strategies are introduced to improve the global optimization ability of the algorithm. To assess the performance of the CSTLBO algorithm, 20 Benchmark functions were tested, and Wilcoxon rank sum tests were applied to the experimental results of six improved TLBO algorithms. The experimental results indicate that the CSTLBO algorithm is able to avoid premature convergence, and has the advantages in global exploration, solution quality, and stability.

Keywords: swarm Intelligence; teaching-learning-based optimization; random crossover; “self-learning” strategy; Benchmark function; non-origin-optimal; diversity analysis

教与学优化算法(teaching-learning-based optimization, TLBO)是 Rao 等^[1]于 2011 年提出的一

种模拟教师教学和学生相互学习的群体智能优化算法。该算法因参数设置少、容易实现、寻优性能好, 受到了众多研究者的关注, 在函数优化^[2]、工程参数优化^[3]、投资组合优化^[4]和多目标优化^[5-6]等问题上得到了广泛的应用。

TLBO 算法作为一种群智能优化算法, 在求

收稿日期: 2019-10-31. 网络出版日期: 2020-07-21.

基金项目: 国家自然科学基金项目(11502132); 陕西省教育厅重点科研计划项目(20JS021); 陕西理工大学校级科研项目(SLG1913).

通信作者: 黎延海. E-mail: liyanhai@snut.edu.cn.

解高维多模态复杂优化问题时也容易陷入局部最优,算法的收敛速度和精度都不够理想,为此,许多研究者在近几年相继提出了一些改进的 TLBO 算法。Ouyang 等^[7]为平衡全局与局部搜索效率,在“学阶段”增加了全局交叉策略,提出了全局交叉的 TLBO 算法 (teaching-learning-based optimization with global crossover, TLBO-GC); Chen 等^[8]设计了局部学习和自学习方法来提高 TLBO 的搜索能力,提出了一种改进的 TLBO 算法 (improved teaching-learning-based optimization, ITLBO); Zou 等^[9]利用其他同学的学习经验,使用两种随机概率来确定学习者在不同阶段的学习方法,提出了一种改进的 TLBO 算法 (improved teaching-learning-based optimization with learning experience, LETLBO); 王培崇等^[10]采用小世界网络作为种群的空间结构关系,提出了具有小世界领域结构的 TLBO 算法 (small world neighborhood TLBO, S-TLBO); 毕晓君等^[11]在“学阶段”融合差分进化算法变异策略,提出了基于混合学习策略的教与学优化算法 (TLBO based on hybrid learning strategies and disturbance, DSTLBO); Ji 等^[12]受历史种群概念的启发,在标准 TLBO 算法中引入了自反馈学习阶段和变异交叉阶段,提出了一种 TLBO 算法的变体 (improved version of TLBO, I-TLBO); Yu 等^[13]将精英学习策略和多样性学习方法引入到教师阶段和学习者阶段,学习者可以根据自身的特点自适应地选择不同的学习阶段,提出了一种自适应的 TLBO 算法 (self-adaptive TLBO, SATLBO); Niu 等^[14]对种群中每个个体采用不同的更新机制,提出了一种修正的 TLBO 算法 (actual teaching learning situation TLBO, ATLS); SHUKLA 等^[15]在标准 TLBO 算法中引入邻近学习和差分突变策略,提出了邻近学习策略的 TLBO 算法 (neighbor based TLBO, NTLBO); 柳缔西子等^[16]利用权重学习得到的个体来指引种群的进化,使用 Logistics 混沌搜索策略来提高其全局搜索能力,提出了基于混沌搜索和权重学习的教与学优化算法 (TLBO based on chaotic search and weighted learning, TLBO-CSWL); 何杰光等^[17]在原有的个体更新公式上加入更新个体局部维度的操作,提出了基于个体局部维度改进的教与学优化算法 (local-dimension-improved TLBO, LdimTLBO); Tsai^[18]将交叉频率引入 TLBO 中,以防止算法过早收敛,提出了受限的教与学优化算法 (confined TLBO with variable search strategies, CTLBO)。这些 TLBO 的改进算法在一定程度上提高了算法的优化性能,降低了算法陷入局部最优的可能,但文献^[19]揭示标准 TLBO 算法在“教”阶段存在固

有的起源偏倚,对原点最优问题 (以坐标原点为最优解的问题) 有着原始的偏好,但对非原点最优问题 (最优解不在坐标原点的问题) 的优化效果则不太理想,现有的改进 TLBO 算法也没能很好地解决这类问题。

为此,本文在对标准 TLBO 算法的“教阶段”和“学阶段”的空间扰动进行几何分析解释的基础上,提出了一种基于随机交叉-自学策略的教与学优化算法 (teaching and learning optimization algorithm based on random crossover-self-study strategy, CSTLBO)。算法使用各分量取值 $[-1, 1]$ 的随机向量替换了标准 TLBO 算法中各分量取值 $[0, 1]$ 的随机向量,通过引入随机交叉策略和“自学”策略,增加种群的多样性,提高算法的全局搜索能力。使用了 20 个 Benchmark 测试函数进行仿真实验,并与 6 种 TLBO 的改进算法 (TLBO-GC^[7]、ITLBO^[8]、DSTLBO^[11]、ATLS^[14]、NTLBO^[15]、TLBO-CSWL^[16]) 进行了比较,检验了所提算法的性能。

1 教与学优化算法

1.1 优化问题的模型

本文研究的优化问题的具体模型为:

$$\begin{aligned} & \min_{\mathbf{X}} f(\mathbf{X}) \\ & \text{s.t. } \mathbf{X} = (x_1, x_2, \dots, x_D) \in \mathbf{S}, \\ & x_i \in [x_i^L, x_i^U], i = 1, 2, \dots, D \end{aligned}$$

式中: \mathbf{X} 为决策向量; \mathbf{S} 表示搜索空间; x_i^L 和 x_i^U 分别表示第 i 个决策变量 x_i 在搜索空间的上界和下界; D 是搜索空间的维数; f 为目标函数。

1.2 标准的教与学优化算法

标准 TLBO 算法是一种模仿整个课堂教学过程的高效优化方法,通过模拟老师给班级学生的教学过程及班级中学员之间的交流学习过程,来提高整个班级的成绩。主要包括两个阶段:“教阶段”和“学阶段”,其基本步骤如下:

1) 初始化。随机初始化 NP 位学员 (即个体) $\mathbf{X}^j = (x_1^j, x_2^j, \dots, x_D^j)$, $j = 1, 2, \dots, \text{NP}$ 。其中, NP 表示班级中的学员数 (即种群的规模), D 表示学习的科目数 (即待优化问题的维数)。

2) “教阶段”。选取班级中成绩最优 (即适应度值最小) 的学员作为老师,记做 $\mathbf{X}^{\text{teacher}}$, 每个学员根据教师和全体学员平均水平的差异性 (difference) 来进行学习,具体如下:

$$\mathbf{X}_{\text{new}}^j = \mathbf{X}_{\text{old}}^j + \text{difference} \quad (1)$$

$$\text{difference} = \mathbf{r} \otimes (\mathbf{X}^{\text{teacher}} - \mathbf{T}_F \cdot \mathbf{X}^{\text{mean}}) \quad (2)$$

式中: $\mathbf{X}_{\text{old}}^j$ 、 $\mathbf{X}_{\text{new}}^j$ 分别表示第 j 位学员 \mathbf{X}^j 在教学前、后的知识水平; $\mathbf{r} = (r_1, r_2, \dots, r_D)$ 表示学习步长, r_i 为 $[0, 1]$ 间的随机数; 符号 \otimes 表示直积运算;

$T_F = \text{round}[1 + \text{rand}(0, 1)]$ 表示教学因子, 取值为 1 或 2; $X^{\text{mean}} = \frac{1}{NP} \sum_{j=1}^{NP} X^j$ 表示班级全体学员的平均知识水平。

“教阶段”完成后, 评估学员 X^j 的学习成绩 (即适应度值 $f(X^j)$), 如果 $f(X_{\text{new}}^j) < f(X_{\text{old}}^j)$, 则进行更新, 即 $X^j = X_{\text{new}}^j$, 否则不更新。

3) “学阶段”。学员之间进行相互交流, 通过向其他同学学习, 来提升自己的水平。从班级中随机抽取一个学员 X^k , k 是 $[1, NP]$ 中的一个随机整数, $k \neq j$, 学员 X^j 与其进行交流学习, 具体如下:

$$X_{\text{new}}^j = \begin{cases} X_{\text{old}}^j + r \otimes (X^k - X^j), & \text{if } f(X^k) < f(X^j) \\ X_{\text{old}}^j + r \otimes (X^j - X^k), & \text{其他} \end{cases} \quad (3)$$

学习完成后, 如果 $f(X^k) < f(X^j)$, 则更新个体, 即 $X^j = X_{\text{new}}^j$, 否则不更新。

4) 如果满足终止条件 (达到最大评估次数或最大迭代次数), 则优化结束并输出最优个体 X_{best} , 否则转至 2) 继续。

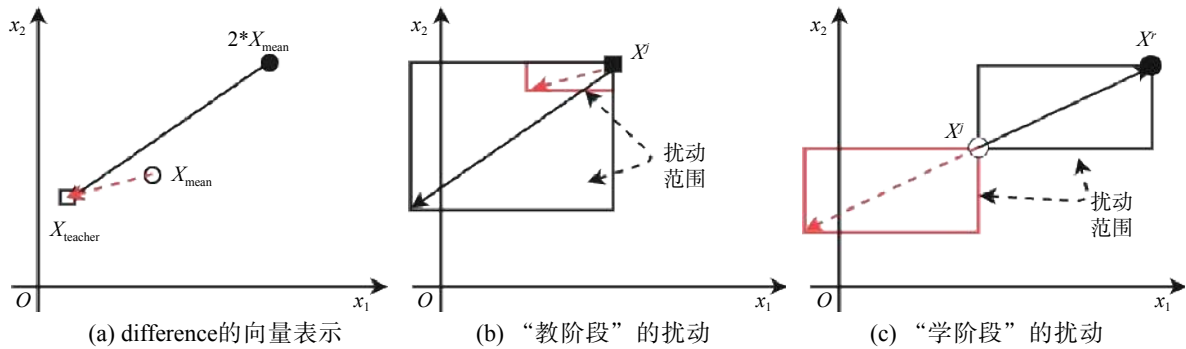


图1 扰动分析的几何解释

Fig. 1 Geometric interpretation of disturbance analysis

2.2 随机向量的扰动分析

类似于 2.1 节的分析, 如果将式 (2) 和式 (3) 中各分量取值于 $[0, 1]$ 的随机向量替换为各分量取值于 $[-1, 1]$ 的随机向量, 则应用于 X^j 的扰动范围将会扩大, “教阶段”的搜索扰动范围会变为原来的 4 倍, 而“学阶段”的扰动范围将变为原来的两

2 教与学优化算法的空间扰动分析

2.1 标准 TLBO 算法的几何解释

为了更好地理解 TLBO 的工作原理, 以二维空间为例, 采用几何分析的方法来解释算法在“教阶段”和“学阶段”的空间扰动状况。

文献 [19] 指出式 (2) 和 (3) 中的 r 本质上是一个各分量都取值于 $[0, 1]$ 的 D 维随机向量, 考虑到由随机参数所引起的扰动的极限情形, 容易观察到“教阶段”和“学阶段”的空间扰动情形。由于 r 中的每个元素都是取值于 $[0, 1]$ 的随机数, 所以 r 对 difference 的影响实际是一个尺度上的缩放, 即 difference 可以为图 1(b)、(c) 方框内的任意向量。图 1(a) 给出了 T_F (取 1 或 2) 的两种情况下 difference 的向量表示; 图 1(b) 描述了“教阶段”, 应用于 X^j 的两个可能的扰动框, 从概率角度描述了应用于 X^j 每一个可能的扰动; 图 1(c) 描述了“学阶段”, 应用于 X^j 的两种可能的扰动框。

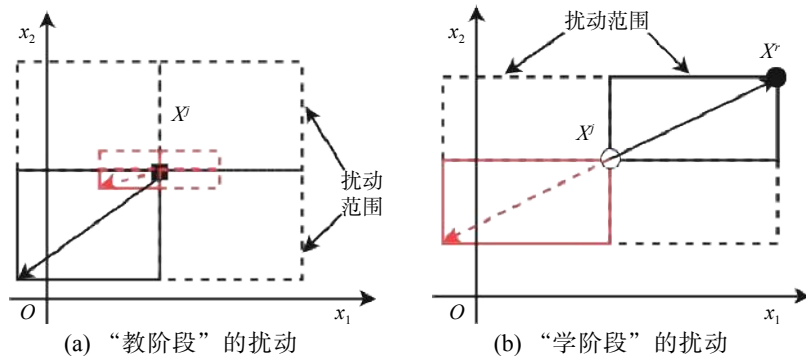


图2 扰动分析的几何解释 ($r_i \in [-1, 1]$)

Fig. 2 Geometric interpretation of disturbance analysis ($r_i \in [-1, 1]$)

倍, 这样能增大算法跳出局部最优的概率, 在一定程度上改善算法“早熟”收敛的状况, 提高全局搜索能力。

图 2(a)、(b) 分别描述了在“教阶段”和“学阶段”, 使用各分量取值于 $[-1, 1]$ 的随机向量后, 应用于 X^j 的可能的扰动范围。

2.3 随机扰动的实验分析

为检验上述对使用不同取值范围的随机数所产生的扰动状况的分析,将标准 TLBO 算法和 $TLBO_1(r)$ (各分量取值于 $[-1,1]$ 的随机向量) 在 6 个标准测试函数上进行仿真测试,其中 $f_1 \sim f_3$ 是

典型的原点最优的多峰问题, $f_4 \sim f_6$ 是对它们进行了 Shift 操作 (非原点最优)。设置维度 $D=30$, 种群大小 $NP=10$, 评价次数为 150 000 次。表 1 统计了 30 次独立运行得到的最优目标值的平均值、标准差及运行时间,图 3 给出 6 个函数的收敛曲线。

表 1 30 次实验统计结果

Table 1 Thirty times optimization statistical results

| 函数 | 算法 | 平均值 \pm 标准差 | 时间/s |
|----------------------------------|----------|---|------|
| f_1 : Ackley Function | TLBO | $5.32 \times 10^{-15} \pm 1.58 \times 10^{-15}$ | 1.22 |
| | $TLBO_1$ | $7.99 \times 10^{-15} \pm 0.00$ | 1.16 |
| f_2 : Griewank Function | TLBO | 0.00 ± 0.00 | 1.22 |
| | $TLBO_1$ | $1.48 \times 10^{-03} \pm 3.69 \times 10^{-03}$ | 1.33 |
| f_3 : Rastrigin Function | TLBO | $1.97 \times 10^{+01} \pm 1.30 \times 10^{+01}$ | 1.04 |
| | $TLBO_1$ | $5.76 \times 10^{+01} \pm 2.98 \times 10^{+01}$ | 1.05 |
| f_4 : Ackley Shift Function | TLBO | $1.82 \times 10^{+01} \pm 1.04$ | 4.31 |
| | $TLBO_1$ | $7.64 \times 10^{-15} \pm 2.09 \times 10^{-15}$ | 4.14 |
| f_5 : Griewank Shift Function | TLBO | 2.29 ± 4.47 | 4.33 |
| | $TLBO_1$ | $8.63 \times 10^{-04} \pm 2.69 \times 10^{-03}$ | 4.12 |
| f_6 : Rastrigin Shift Function | TLBO | $2.05 \times 10^{+02} \pm 2.82 \times 10^{+01}$ | 3.11 |
| | $TLBO_1$ | $4.67 \times 10^{+01} \pm 2.68 \times 10^{+01}$ | 2.99 |

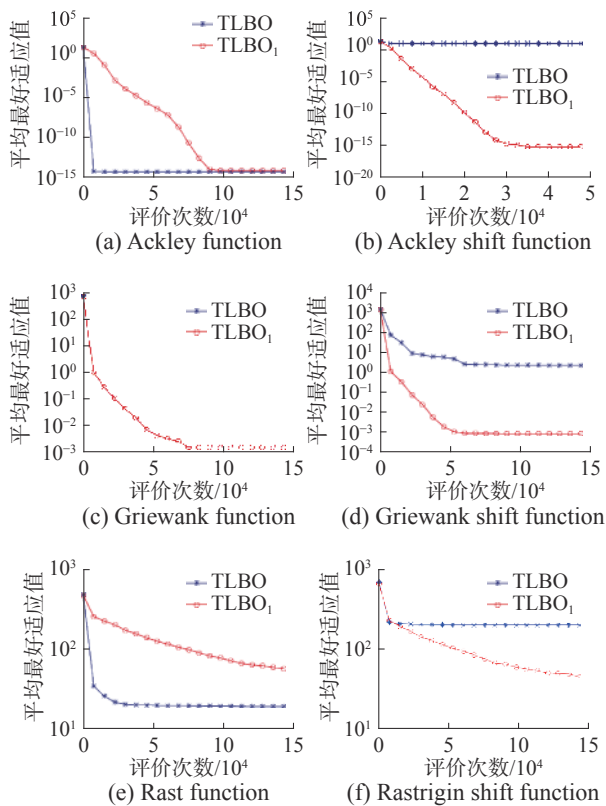


图 3 测试函数 $f_1 \sim f_6$ 的收敛曲线图

Fig. 3 Convergence plot of benchmark function $f_1 \sim f_6$

从表 1 和图 3 可以看出,对原点最优问题,标准 TLBO 算法所得解的精度更高,时间更短,但对非原点最优问题, $TLBO_1$ 算法的求解精度和时间则更优。由于标准 TLBO 算法的“教”阶段存在固

有的原点偏倚,允许收敛的种群继续沿着原点的方向搜索更好的解决方案,从而对原点最优问题能取得更好的优化效果。对非原点最优问题,标准 TLBO 算法在种群收敛到局部最优时,要么继续利用局部最优信息,要么沿着原点方向继续探索,由于最优解在原点以外的点上,沿原点方向的搜索几乎毫无意义,所以无法进一步获得高精度的解;而采用各分量取值于 $[-1,1]$ 的随机向量,使得搜索范围成倍扩大,算法跳出局部最优的能力得到提高,获得最优解的概率变大,随着迭代的进行,能取得更好的优化效果。

3 基于随机交叉-自学策略的教与学优化算法

标准 TLBO 算法中,由于教师和学习对象都属于现有种群,所以无论是“教阶段”还是“学阶段”的操作都是在当前位置的有限范围内进行搜索,学习者都依据现有种群的经验来提高自己的成绩,使得算法具有良好的开发性能,而跳出局部最优的能力不足。为增强算法的全局搜索能力,使迭代过程中算法的探索与开发过程得到平衡,对“教”和“学”阶段进行改进,并使用随机交叉策略和“自学”策略,来提高算法对复杂多模态问题的优化性能。

3.1 “教阶段”的改进

当班级中学生成绩差异较大时,最差成绩往往会拉低班级的平均水平,如果教学过程中重点

加强对最差生的关注,使其成绩得到大幅度提高,缩小与班级其他同学间的差距,则可以加快班级整体水平的提升。故用班级中的成绩最差者 X^{worst} 替换班级平均水平 X^{mean} , 即

$$\text{difference} = r_1 \otimes (X^{\text{teacher}} - T_F \cdot X^{\text{worst}}) \quad (4)$$

$$X_{\text{new}}^j = \begin{cases} X_{\text{old}}^j + r_2 \otimes \left[\left(1 - \frac{t}{T_{\max}}\right) \cdot X_{\text{old}}^k + \left(\frac{t}{T_{\max}}\right) \cdot X^{\text{teacher}} - X_{\text{old}}^j \right], & \text{if } f(X^k) < f(X^j) \\ X_{\text{old}}^j + r_2 \otimes \left[\left(1 - \frac{t}{T_{\max}}\right) \cdot X_{\text{old}}^j + \left(\frac{t}{T_{\max}}\right) \cdot X^{\text{teacher}} - X_{\text{old}}^k \right], & \text{其他} \end{cases}$$

其中, r_2 为各分量取值于 $[-1, 1]$ 的随机向量, k 是 $[1, \text{NP}]$ 中的一个随机整数, $k \neq j$, t 为当前迭代次数, T_{\max} 为最大迭代次数。在迭代初期, t 较小, $(1 - t/T_{\max})$ 的值较大, 学生间的相互讨论更多, 算法更加注重随机性学习, 能够维持种群多样性, 增大有效搜索范围; 随着迭代的进行, t 逐渐增大, (t/T_{\max}) 越来越大, 学生会更多地向老师求教, 逐渐侧重于有向性学习, 使算法的局部搜索能力增强。

3.3 随机交叉-自学策略

3.3.1 随机交叉策略

学生经过“教”阶段后, 各自的成绩会得到一定程度的提升, 但彼此间的差距也会缩小, 即种群的多样性降低, 而通过集体讨论交流, 每位学生都可以在不同科目上随机地向班级中不同的同学求教, 并达到该同学在此科目上的水平, 这实际上就是种群间的一种随机交叉过程, 具体为

$$x_{\text{new},i}^j = \begin{cases} x_{\text{old},i}^k, & \text{rand} < \text{rand} \\ x_{\text{old},i}^j, & \text{其他} \end{cases} \quad (6)$$

$i = 1, 2, \dots, D, j = 1, 2, \dots, \text{NP}$, k 是 $[1, \text{NP}]$ 中的一个不同于 j 的随机整数, rand 为 $[0, 1]$ 间的随机数。这里使用一种随机概率来确定学习者与不同学习对象间的交流程度, 通过进行种群间的随机交叉, 能够维持种群的多样性, 提高算法跳出局部最优的能力。

3.3.2 “自学”策略

实际学习过程中, 部分学生在学习之余, 也会根据自己当前的学习状况, 挑选部分课程进行“自学”, 发掘新的知识, 从而提升自己的学业水平。具体如下: 对第 i 门课程, 产生一个 $[0, 1]$ 间的随机数 rand , 来用于选择不同的“自学”方式。

$$x_{\text{new},i}^j = \begin{cases} x_{\text{old},i}^j + 2 \cdot (\text{rand} - 0.5) \cdot \lambda, & \text{rand} < \text{rand} \\ x_i^L + \text{rand} \cdot (x_i^U - x_i^L), & \text{rand} < \text{SDR} \\ x_{\text{old},i}^j, & \text{其他} \end{cases} \quad (7)$$

$$\lambda = \lambda_{\max} \times \left(\frac{\lambda_{\min}}{\lambda_{\max}} \right)^{\left(\frac{t}{T_{\max}} \right)^2} \quad (8)$$

$i = 1, 2, \dots, D, j = 1, 2, \dots, \text{NP}$ 。其中, SDR (self-development rate, SDR) 为自我拓展率, λ 表示自学步长, λ_{\max} 、 λ_{\min} 分别为最大、最小步长。

对每一科目, 学生会根据自己当前的知识水平

式中 r_1 为各分量取值于 $[-1, 1]$ 的随机向量。

3.2 “学阶段”的改进

学生学习时, 不仅会在同学间进行相互交流, 也会向老师请教。在迭代的前期, 同学间的信息交流较多, 但随着迭代的进行, 学生会更多地向老师请教, 故对学习方式做如下改进:

$$\text{其他} \quad (5)$$

随机进行自我调整, 提高学习效率, 自学步长随迭代的进行由 λ_{\max} 逐步动态缩小为 λ_{\min} 。搜索前期, 步长值较大, 算法具有强的空间扰动能力, 有助于探索新的未知区域; 随着迭代的进行, 步长值逐渐缩小, 使得对当前解的扰动幅度变小, 有利于提高解的精度, 这样一种动态调整过程能有效平衡算法在搜索过程中的空间扰动能力和局部寻优能力。同时, 以概率 SDR 在搜索空间进行随机搜寻, 进行自我拓展学习, 可以增加算法的全局搜索能力。大量数值实验测试表明, 自我拓展率取值不宜过大, 当 $\text{SDR} \in [0.01, 0.1]$ 时, 算法会有较好的性能。

3.4 CSTLBO 算法的流程

CSTLBO 算法的具体流程如图 4 所示。

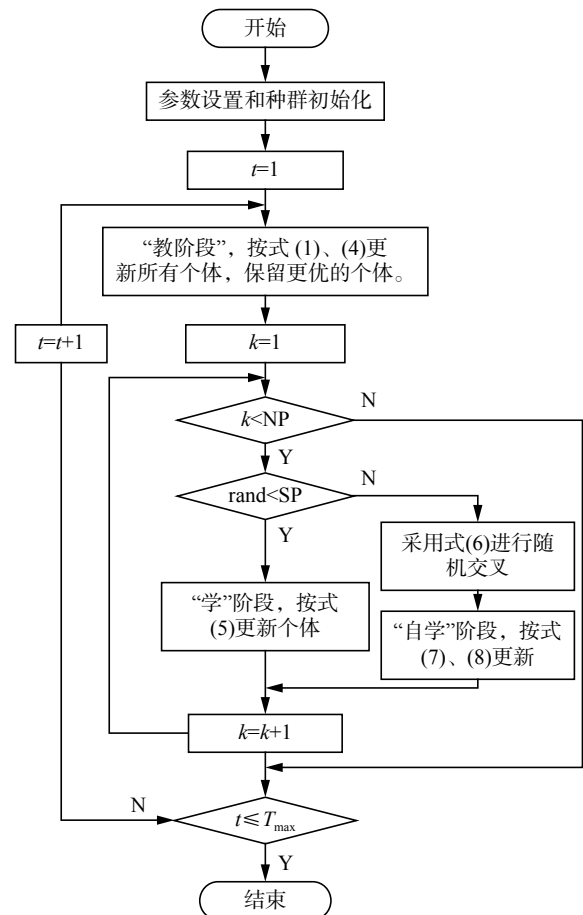


图 4 CSTLBO 算法流程

Fig. 4 Flow chart of CSTLBO algorithm

每次迭代过程中,对每位学生而言,依据选择概率SP来选择“学”和“自学”两种学习方式中的一种执行,具体如下:

$$SP = SP_{\max} - \left(SP_{\max} - SP_{\min} \right) \times \left(\frac{t}{T_{\max}} \right)^{(1/2)} \quad (9)$$

随迭代的进行,SP的值从 SP_{\max} 逐渐减小到 SP_{\min} ,表明在迭代初期,同学之间的信息交流比较多,随着学习的深入,当学生的水平都得到较大提升后,会更多地倾向于进行自我探索学习。

4 实验仿真测试

4.1 测试函数与参数设置

为检验 CSTLBO 算法在连续优化问题中的性能,将其与 6 种 TLBO 的改进算法 (ATLS、NTLBO、TLBO-GC、ITLBO、TLBO-CSWL、DSTLBO) 在 20 个复杂 Benchmark 函数上进行测试比较。将函数 $F_1, F_3 \sim F_4$ 称为原点最优问题(它们在原点处得到了最优值),其余的函数称为非原点最优问题,其中一些是通过对原点最优问题进行移位和旋转得到的,移位函数取自 CEC 2008,移位和旋转函数取自 CEC 2017。

F_1 为 Ackley Function; F_2 为 Michalewics Function; F_3 为 Rastrigin Function; F_4 为 Griewank Function; F_5 为 Schwefel 2.26 Function; F_6 为 Weierstrass Function; F_7 为 FastFractal ‘DoubleDip’ Function; F_8 为 Sphere Shift Function; F_9 为 Ackley Shift Function; F_{10} 为 Griewank Shift Function; F_{11} 为 Rastrigin Shift Function; F_{12} 为 Rosebrock shift Func-

tion; F_{13} 为 Schaffer Shift Function; F_{14} 为 Bohachevsky Shift Function; F_{15} 为 Shifted and Rotated Rastrigin’s Function; F_{16} 为 Shifted and Rotated Expanded Scaffer’s F_6 Function; F_{17} 为 Shifted and Rotated Lunacek Bi_Rastrigin Function; F_{18} 为 Shifted and Rotated Non-Continuous Rastrigin’s Function; F_{19} 为 Shifted and Rotated Levy Function; F_{20} 为 Shifted and Rotated Schwefel’s Function.

本文测试的环境为:戴尔 PC 机 Intel Core (TM)i7-4790 3.6 GHz CPU, 8 GB 内存; Window10 操作系统; MATLAB R2014b 软件。为保证测试的公平,各种算法采用相同的适应度函数评价次数:对 $F_1 \sim F_{14}$, 评价次数 $\text{MaxFEs} = 5\,000 \times D$; 对 $F_{15} \sim F_{20}$, $\text{MaxFEs} = 10\,000 \times D$, 其中 D 为维数。比较算法的参数设置均参考于原文献,本文算法的参数设置为: $\text{NP} = 20$, $\text{SP}_{\max} = 0.6$, $\text{SP}_{\min} = 0.2$, $\text{SDR} = 0.02$, $\lambda_{\max} = (xU - xL)/10$, $\lambda_{\min} = (xU - xL)/(10^{15})$ 。

4.2 实验结果与分析

为避免偶然因素的影响,所有算法在 $D=100$ 时都独立运行了 30 次,表 2 记录了各种算法在 20 个测试函数上 30 次独立实验的统计结果,对 $F_1 \sim F_{14}$, 记录了最优目标值的平均值和标准差,对 $F_{15} \sim F_{20}$, 记录了最优误差的平均值和标准差。为检验本文算法与比较算法之间的差异性,进行了显著性水平为 0.05 的 Wilcoxon 秩和检验,“+”、“-”、“ \approx ”分别表示比较算法与本文算法的成绩相比是好、差和相当。

表 2 算法 30 次独立测试结果 ($D=100$)

Table 2 Thirty times optimization results of the algorithm ($D=100$)

| | ATLS | NTLBO | TLBO-GC | ITLBO | TLBO-CSWL | DSTLBO | CSTLBO |
|-------|----------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|
| F_1 | 平均值 8.88×10^{-16} | 3.20×10^{-15} | 7.99×10^{-15} | 6.57×10^{-15} | 1.95×10^{-15} | 7.46×10^{-15} | 5.93×10^{-13} |
| | 标准差 0.00 | 1.74×10^{-15} | 0.00 | 1.79×10^{-15} | 1.67×10^{-15} | 1.30×10^{-15} | 3.28×10^{-14} |
| F_2 | 平均值 -2.15×10^1 | -7.96×10^1 | -7.89×10^1 | -7.69×10^1 | -3.48×10^1 | -8.90×10^1 | -9.93×10^1 |
| | 标准差 1.37 | 5.31 | 3.63 | 3.13 | 2.04 | 2.97 | 6.19×10^{-2} |
| F_3 | 平均值 0.00 | 0.00 | 1.63×10^2 | 0.00 | 0.00 | 4.63 | 5.68×10^{-15} |
| | 标准差 0.00 | 0.00 | 4.45×10^1 | 0.00 | 0.00 | 1.42×10^1 | 2.54×10^{-14} |
| F_4 | 平均值 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| | 标准差 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| F_5 | 平均值 3.76×10^4 | 1.68×10^4 | 1.24×10^4 | 1.65×10^4 | 1.14×10^4 | 9.67×10^3 | 1.31×10^{-10} |
| | 标准差 1.04×10^3 | 2.50×10^3 | 1.27×10^3 | 2.27×10^3 | 5.05×10^1 | 1.10×10^3 | 0.00 |
| F_6 | 平均值 1.84×10^2 | 1.52×10^2 | 8.88×10^1 | 1.19×10^2 | 1.47×10^2 | 9.71×10^1 | 1.75×10^{-11} |
| | 标准差 3.56 | 4.76 | 3.71 | 7.57 | 3.06 | 1.13×10^1 | 4.55×10^{-11} |
| F_7 | 平均值 -7.64×10^2 | -1.15×10^3 | -1.30×10^3 | -1.26×10^3 | -8.96×10^2 | -1.20×10^3 | -1.54×10^3 |
| | 标准差 3.04×10^1 | 3.59×10^1 | 3.35×10^1 | 4.27×10^1 | 2.61×10^1 | 2.90×10^1 | 4.14 |

续表 2

| | | ATLS | NTLBO | TLBO-GC | ITLBO | TLBO-CSWL | DSTLBO | CSTLBO |
|----------|-----|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|------------------------|
| F_8 | 平均值 | 3.47×10^5 | 4.35×10^2 | 8.91×10^3 | 2.20×10^{-3} | 3.04×10^5 | 2.36×10^{-9} | 2.28×10^{-23} |
| | 标准差 | 5.98×10^3 | 1.38×10^3 | 4.21×10^3 | 7.81×10^{-3} | 1.59×10^4 | 1.05×10^{-8} | 2.22×10^{-24} |
| F_9 | 平均值 | 2.09×10^1 | 1.93×10^1 | 1.80×10^1 | 1.86×10^1 | 2.06×10^1 | 1.71×10^1 | 6.19×10^{-13} |
| | 标准差 | 4.44×10^{-2} | 7.41×10^{-2} | 6.07×10^{-1} | 2.75×10^{-1} | 7.70×10^{-2} | 6.68×10^{-1} | 2.81×10^{-14} |
| F_{10} | 平均值 | 2.76×10^3 | 3.63 | 8.61×10^1 | 1.29×10^{-1} | 2.35×10^3 | 1.50 | 1.11×10^{-17} |
| | 标准差 | 5.50×10^1 | 5.36 | 5.33×10^1 | 1.64×10^{-1} | 1.93×10^2 | 3.80 | 3.42×10^{-17} |
| F_{11} | 平均值 | 1.85×10^3 | 9.32×10^2 | 5.96×10^2 | 7.32×10^2 | 1.60×10^3 | 6.82×10^2 | 7.50×10^{-14} |
| | 标准差 | 6.45×10^1 | 4.98×10^1 | 8.11×10^1 | 5.52×10^1 | 5.39×10^1 | 5.71×10^1 | 1.04×10^{-13} |
| F_{12} | 平均值 | 9.37×10^{10} | 2.35×10^7 | 1.15×10^9 | 4.69×10^2 | 7.91×10^{10} | 3.51×10^2 | 2.47×10^2 |
| | 标准差 | 6.70×10^9 | 5.93×10^7 | 1.14×10^9 | 4.89×10^2 | 4.17×10^9 | 3.61×10^2 | 1.20×10^2 |
| F_{13} | 平均值 | 8.63×10^2 | 6.29×10^2 | 4.99×10^2 | 5.54×10^2 | 7.97×10^2 | 5.00×10^2 | 2.54×10^{-1} |
| | 标准差 | 1.49×10^1 | 1.34×10^1 | 3.33×10^1 | 2.67×10^1 | 1.60×10^1 | 2.64×10^1 | 1.86×10^{-1} |
| F_{14} | 平均值 | 4.87×10^3 | 5.61×10^1 | 1.74×10^2 | 3.65×10^1 | 4.38×10^3 | 3.84×10^1 | 0.00 |
| | 标准差 | 1.23×10^2 | 9.44 | 8.26×10^1 | 7.58 | 3.13×10^2 | 5.55 | 0.00 |
| F_{15} | 平均值 | 1.76×10^3 | 8.61×10^2 | 8.90×10^2 | 6.22×10^2 | 1.49×10^3 | 5.78×10^2 | 3.16×10^2 |
| | 标准差 | 6.44×10^1 | 5.44×10^1 | 5.92×10^1 | 7.33×10^1 | 5.22×10^1 | 4.21×10^1 | 4.10×10^1 |
| F_{16} | 平均值 | 1.23×10^2 | 6.39×10^1 | 5.34×10^1 | 4.83×10^1 | 1.02×10^2 | 3.98×10^1 | 2.27×10^{-13} |
| | 标准差 | 4.40 | 2.22 | 5.80 | 3.56 | 5.17 | 5.44 | 0.00 |
| F_{17} | 平均值 | 3.48×10^3 | 2.53×10^3 | 1.80×10^3 | 1.72×10^3 | 3.16×10^3 | 1.36×10^3 | 4.40×10^2 |
| | 标准差 | 1.20×10^2 | 7.10×10^1 | 1.90×10^2 | 1.57×10^2 | 8.93×10^1 | 2.16×10^2 | 4.97×10^1 |
| F_{18} | 平均值 | 1.94×10^3 | 1.03×10^3 | 9.42×10^2 | 7.17×10^2 | 1.66×10^3 | 6.01×10^2 | 2.87×10^2 |
| | 标准差 | 5.29×10^1 | 6.47×10^1 | 6.57×10^1 | 8.63×10^1 | 7.26×10^1 | 5.73×10^1 | 3.26×10^1 |
| F_{19} | 平均值 | 9.89×10^4 | 2.24×10^4 | 2.24×10^4 | 1.58×10^4 | 6.86×10^4 | 1.18×10^4 | 3.48×10^{-2} |
| | 标准差 | 7.07×10^3 | 1.09×10^3 | 3.04×10^3 | 1.85×10^3 | 4.61×10^3 | 1.97×10^3 | 1.56×10^{-1} |
| F_{20} | 平均值 | 3.40×10^4 | 1.65×10^4 | 1.97×10^4 | 1.36×10^4 | 2.91×10^4 | 1.46×10^4 | 2.29×10^4 |
| | 标准差 | 8.27×10^2 | 1.69×10^3 | 1.14×10^3 | 1.41×10^3 | 8.03×10^2 | 1.30×10^3 | 1.09×10^3 |
| 合 计 | + | 2 | 3 | 1 | 3 | 2 | 2 | |
| | - | 17 | 16 | 17 | 16 | 16 | 17 | |
| | ≈ | 1 | 1 | 2 | 1 | 2 | 1 | |

从表 2 可以看出, 所有比较算法在 F_1 这个原点最优问题上的收敛精度优于本文算法, 在 F_4 上所有算法都能达到理论最优值, 除 F_{20} 外, 本文算法在余下的 17 个测试函数上的平均值和方差都优于比较算法, 说明所提算法对非原点最优问题的优化能力较强, 效果良好。从 Wilcoxon 符号秩检验的结果来看, 比较算法最多在 4 个问题上优于或相当于本文算法, 而在其余测试问题上都明显差于本文算法。

为进一步分析实验结果, 图 5~6 给出了 30 次独立运行的平均最优目标值收敛曲线 ($D=100$ 维), 图 7~8 给出了 30 次独立运行的最优目标值分布盒图 ($D=100$ 维)。

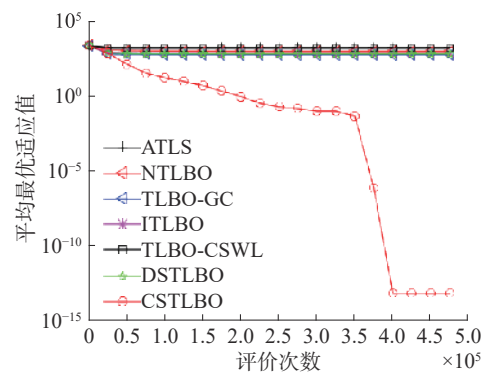
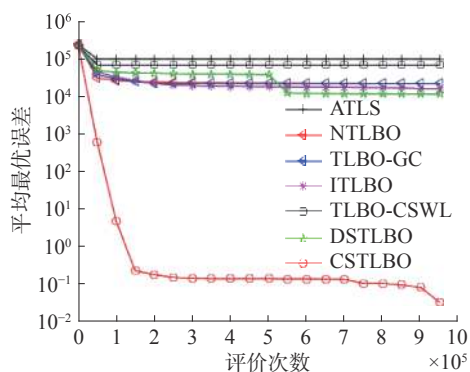
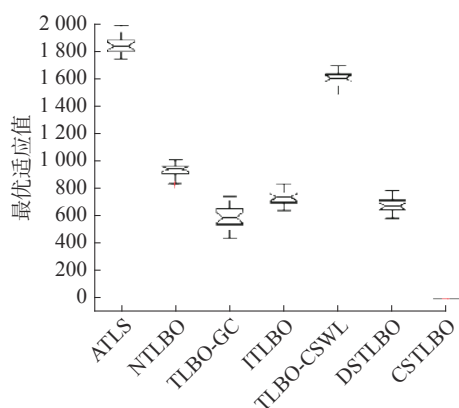
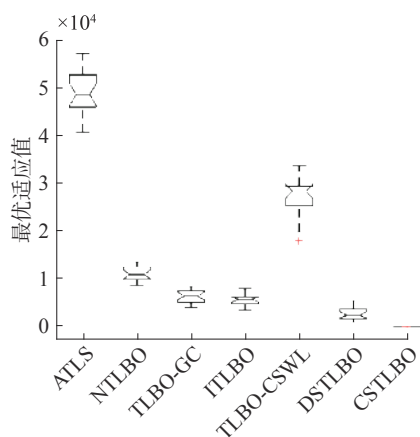


图 5 测试函数 F_{11} 的收敛性曲线
Fig. 5 Convergence plot of benchmark function F_{11}

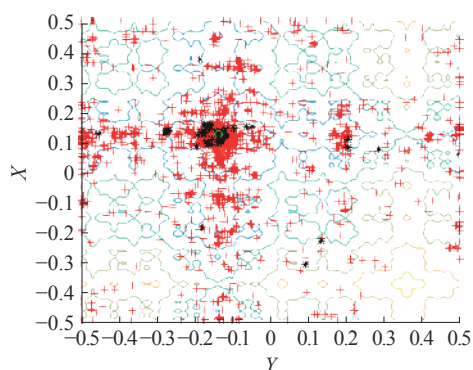
图6 测试函数 F_{19} 的收敛性曲线Fig. 6 Convergence plot of benchmark function F_{19} 图7 测试函数 F_{11} 的最优目标值分布盒图Fig. 7 Boxplot of the optimal value distribution of benchmark function F_{11} 图8 测试函数 F_{19} 的最优目标值分布盒图Fig. 8 Boxplot of the optimal value distribution of benchmark function F_{19}

从收敛曲线图不难看出, 本文算法对两个复杂优化函数 (Rastrigin Shift Function、Shifted and Rotated Levy Function) 的收敛曲线呈逐渐下降趋势, 收敛效果明显优于其他比较算法, 说明本文算法的全局搜索能力较强, 不易陷入局部最优。

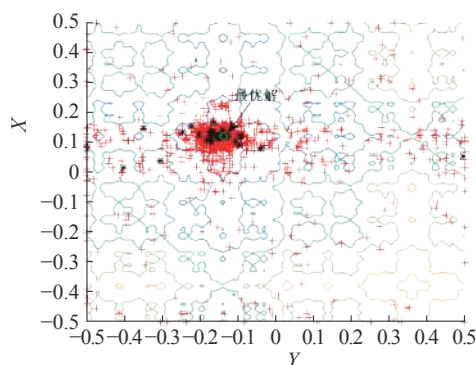
从统计盒图可以看出, 本文算法 30 次独立实验所得的最优值的分布基本呈一条直线, 差距很小, 说明本文算法具有良好的稳定性。

4.3 随机交叉策略的有效性分析

为检验本文所使用的随机交叉策略的有效性, 选取了复杂的非原点最优问题 Weierstrass Function 进行了测试, 函数 Weierstrass Function 具有多个极小值点, 算法很容易陷入局部最优的状况。设置维数 $D=30$, 种群大小 $NP=20$, 最大评价次数 $MaxFEs=150\ 000$, 对搜索过程中产生的所有搜索点和每代的最优搜索点进行了跟踪记录, 用于观察算法的搜索过程。图9描绘了搜索点(取30维中的2维绘制)的分布图。



(a) 随机交叉策略使用前



(b) 随机交叉策略使用后

图9 搜索点分布

Fig. 9 Search point distribution

从图9可以看出, 未使用随机交叉策略的搜索过程中, 种群在多个局部极值处都有聚集, 使得在相同评价次数下, 无法进行有效的开发, 所得解的精度不高; 而使用了随机交叉策略的搜索种群的分布很均匀, 没有发生陷入局部最优的现象, 大多数搜索点都聚集于最优解附近, 更多地来进行局部开发, 说明随机交叉策略能使算法有效地跳出局部最优, 增强算法的全局搜索能力。

4.4 CSTLBO 算法的种群多样性分析

为进一步研究 CSTLBO 算法的收敛过程, 以 4 个复杂非原点最优测试函数 (Schwefel 2.26 Function、Ackley Shift Function、Griewank Shift Function、Rastrigin Shift Function) 为例进行实验 (取维数 $D=100$), 在搜索过程中跟踪记录了算法

种群多样性的变化,如图10所示。种群多样性^[20]定义如下:

$$\text{Diversity} = \frac{1}{D} \sum_{i=1}^D \sqrt{\frac{1}{NP} \sum_{j=1}^{NP} (x_i^j - \bar{x}_i)^2} \quad (10)$$

式中 \bar{x}_i 表示种群中第 i 个决策变量的均值。

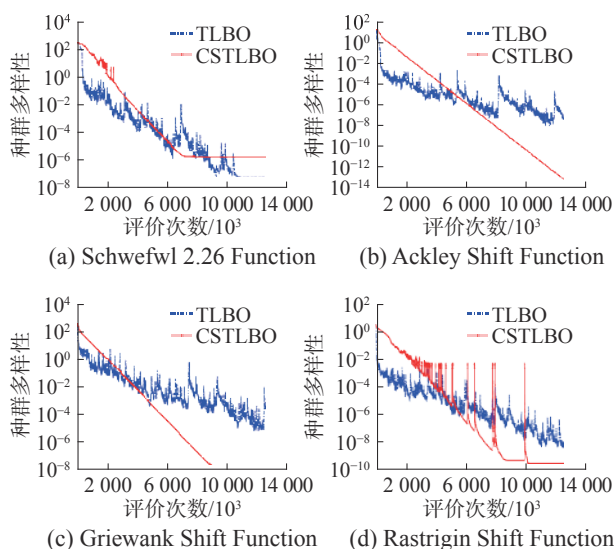


图10 4个复杂非原点最优测试函数的种群多样性曲线
Fig. 10 Population diversity curves of four complex non-origin optimal benchmark function

从图10可以看出,标准TLBO算法在搜索初期多样性明显快速下降,然后在小范围内波动,前后期变化不是特别明显,使得种群个体难以聚集到全局最优解,因此求解精度不高;CSTLBO算法在迭代初期,具有较高的种群多样性,有利于进行全局探索,随着迭代的进行,种群多样性保持着持续下降的趋势,种群逐渐向全局最优解聚集,有利于进行局部开发,从而能获得高精度的全局解。

5 结束语

为提高对非原点最优问题的优化能力,本文提出了一种基于随机交叉-自学策略的教与学优化算法——CSTLBO。算法使用几何方法分析了标准TLBO算法在“教”和“学”阶段的扰动状况,改进了“教”和“学”阶段,并引入了随机交叉策略和“自学”策略来提高算法的全局搜索能力。仿真测试的统计结果表明,所提算法求解精度高,稳定性好,能对较多的非原点最优问题取得良好的优化效果,尤其是对进行了移位操作的复杂问题。但对同时进行了移位和旋转操作的问题,优化结果虽优于比较算法,但在求解精度上仍有提升空间,在后续研究中,可以进一步研究改进,同

时也可以将该算法应用到实际的工程优化问题,检验其具体实用性。

参考文献:

- [1] RAO R V, SAVSANI V J, VAKHARIA D P. Teaching-learning-based optimization: a novel method for constrained mechanical design optimization problems[J]. *Computer-aided design*, 2011, 43(3): 303–315.
- [2] RAO R V, SAVSANI V J, VAKHARIA D P. Teaching-Learning-Based Optimization: an optimization method for continuous non-linear large scale problems[J]. *Information sciences*, 2012, 183(1): 1–15.
- [3] RAO R V, PATEL V. Multi-objective optimization of two stage thermoelectric cooler using a modified teaching-learning-based optimization algorithm[J]. *Engineering applications of artificial intelligence*, 2013, 26(1): 430–445.
- [4] TUO Shouheng, HE Hong. Solving complex cardinality constrained mean-variance portfolio optimization problems using hybrid HS and TLBO algorithm[J]. *Economic Computation and Economic Cybernetics Studies and Research*, 2018, 52(3): 231–248.
- [5] CHINTA S, KOMMADATH R, KOTTECHA P. A note on multi-objective improved teaching-learning based optimization algorithm (MO-ITLBO)[J]. *Information sciences*, 2016, 373: 337–350.
- [6] YU Dong, HONG Jun, ZHANG Jinhua, et al. Multi-objective individualized-instruction teaching-learning-based optimization algorithm[J]. *Applied soft computing*, 2018, 62: 288–314.
- [7] OUYANG Haibin, GAO Liqun, KONG Xiangyong, et al. Teaching-learning based optimization with global crossover for global optimization problems[J]. *Applied mathematics and computation*, 2015, 265: 533–556.
- [8] CHEN Debao, ZOU Feng, LI Zheng, et al. An improved teaching-learning-based optimization algorithm for solving global optimization problem[J]. *Information sciences*, 2015, 297: 171–190.
- [9] ZOU Feng, WANG Lei, HEI Xinhong, et al. Teaching-learning-based optimization with learning experience of other learners and its application[J]. *Applied soft computing*, 2015, 37: 725–736.
- [10] 王培崇, 马玥, 耿明月, 等. 具有小世界邻域结构的教与学优化算法[J]. *计算机科学与探索*, 2016, 10(9): 1341–1350.

WANG Peichong, MA Yue, GENG Mingyue, et al. New teaching-learning-based optimization with neighborhood structure based on small world[J]. *Journal of frontiers of computer science and technology*, 2016, 10(9):

- 1341–1350.
- [11] 毕晓君, 王佳荟. 基于混合学习策略的教与学优化算法[J]. *浙江大学学报(工学版)*, 2017, 51(5): 1024–1031.
BI Xiaojun, WANG Jiahui. Teaching-learning-based optimization algorithm with hybrid learning strategy[J]. *Journal of Zhejiang University (Engineering Science)*, 2017, 51(5): 1024–1031.
- [12] JI Xiaoyuan, YE Hu, ZHOU Jianxin, et al. An improved teaching-learning-based optimization algorithm and its application to a combinatorial optimization problem in foundry industry[J]. *Applied soft computing*, 2017, 57: 504–516.
- [13] YU Kunjie, CHEN Xu, WANG Xin, et al. Parameters identification of photovoltaic models using self-adaptive teaching-learning-based optimization[J]. *Energy conversion and management*, 2017, 145: 233–246.
- [14] NIU Peifeng, MA Yunpeng, YAN Shanshan. A modified teaching-learning-based optimization algorithm for numerical function optimization[J]. *International journal of machine learning and cybernetics*, 2019, 10(6): 1357–1371.
- [15] SHUKLA A K, SINGH P, VARDHAN M. Neighbour teaching learning based optimization for global optimization problems[J]. *Journal of intelligent & fuzzy systems*, 2018, 34(3): 1583–1594.
- [16] 柳缔西子, 范勤勤, 胡志华. 基于混沌搜索和权重学习的教与学优化算法及其应用[J]. *智能系统学报*, 2018, 13(5): 818–828.
LIU Dixizi, FAN Qinqin, HU Zhihua. Teaching-learning-based optimization algorithm based on chaotic search and weighted learning and its application[J]. *CAAI transactions on intelligent systems*, 2018, 13(5): 818–828.
- [17] 何杰光, 彭志平, 崔得龙, 等. 局部维度改进的教与学优化算法[J]. *浙江大学学报(工学版)*, 2018, 52(11): 2159–2170.
HE Jieguang, PENG Zhiping, CUI Delong, et al. Teaching-learning-based optimization algorithm with local dimension improvement[J]. *Journal of Zhejiang University (Engineering Science)*, 2018, 52(11): 2159–2170.
- [18] TSAI H C. Confined teaching-learning-based optimization with variable search strategies for continuous optimization[J]. *Information sciences*, 2019, 500: 34–47.
- [19] PICKARD J K, CARRETERO J A, BHAVSAR V C. On the convergence and origin bias of the Teaching-Learning-Based-Optimization algorithm[J]. *Applied soft computing*, 2016, 46: 115–127.
- [20] TUO Shouheng, ZHANG Junying, YONG Longquan, et al. A harmony search algorithm for high-dimensional multimodal optimization problems[J]. *Digital signal processing*, 2015, 46: 151–163.

作者简介:



黎延海, 讲师, 主要研究方向为智能优化算法及应用。



雍龙泉, 教授, 博士, 主要研究方向为优化理论与算法设计、智能优化算法。



拓守恒, 副教授, 博士, CCF 会员, 主要研究方向为智能优化算法、生物信息分析与处理。发表学术论文 40 余篇。