

基于反馈的精英教学优化算法

于坤杰¹ 王 昕² 王振雷¹

摘 要 精英教学优化算法 (Elitist teaching-learning-based optimization, ETLBO) 是一种基于实际班级教学过程的新型优化算法. 本文针对 ETLBO 算法寻优精度低、稳定性差的问题, 提出了反馈精英教学优化算法 (Feedback ETLBO). 在 ETLBO 算法的基础上, 通过在学生阶段之后加入反馈阶段, 增加了学生的学习方式, 保持学生的多样性特性, 提高算法的全局搜索能力. 同时, 反馈阶段是选举成绩较差的学生与教师交流, 使成绩较差的学生快速向教师靠拢, 使算法进行局部精细搜索, 提高算法的寻优精度. 对 6 个无约束及 5 个约束标准函数的测试结果表明, FETLBO 算法与其他算法相比在寻优精度和稳定性上更具优势. 最后将 FETLBO 算法应用于拉压弹簧优化设计问题及 0-1 背包问题, 取得了满意结果.

关键词 进化算法, 精英教学优化算法, 反馈, 函数优化

引用格式 于坤杰, 王昕, 王振雷. 基于反馈的精英教学优化算法. 自动化学报, 2014, 40(9): 1976–1983

DOI 10.3724/SP.J.1004.2014.01976

Elitist Teaching-learning-based Optimization Algorithm Based on Feedback

YU Kun-Jie¹ WANG Xin² WANG Zhen-Lei¹

Abstract Elitist teaching-learning-based optimization (ETLBO) is a novel optimization algorithm based on the practical teaching-learning process of the class. In this paper, we propose a feedback elitist teaching-learning-based optimization (FETLBO) to solve the problem of low precision and poor stability of the ETLBO. Based on the ETLBO, a feedback phase is introduced at the end of the learner phase to increase the learning style and ensure the diversity of students so as to improve the algorithm's global search ability. Meanwhile, the feedback phase is for the slow students to communicate with the teacher and enables them to be close to the teacher quickly, so that the algorithm uses the fine local search and improves the precision. Six unconstrained and five constrained classic tests show that the FETLBO algorithm outperforms the other algorithms in precision and stability. Finally, the FETLBO algorithm is applied to the tension/compression spring design problem and the 0-1 knapsack problem, and obtains satisfactory results.

Key words Evolutionary algorithms, elitist teaching-learning-based optimization algorithm, feedback, function optimization

Citation Yu Kun-Jie, Wang Xin, Wang Zhen-Lei. Elitist teaching-learning-based optimization algorithm based on feedback. *Acta Automatica Sinica*, 2014, 40(9): 1976–1983

收稿日期 2013-07-08 录用日期 2014-02-26

Manuscript received July 8, 2013; accepted February 26, 2014
国家重点基础研究发展计划 (973 计划) (2012CB720500), 国家自然科学基金 (61333010, 21276078, 21206037), 中央高校基本科研业务费专项资金 (863 计划) (2013AA0400701), 上海市科技攻关 (12dz1125100), 十二五国家科技支撑计划 (2012BAF05B00), 上海市重点学科建设项目 (B504), 上海市自然科学基金 (14ZR1421800), 流程工业综合自动化国家重点实验室开放课题基金资助项目 (PAL-N201404) 资助

Supported by National Basic Research Program of China (973 Program) (2012CB720500), National Natural Science Foundation of China (61333010, 21276078, 21206037), The Central University Basic Scientific Research Business Expenses Special Funds (863 Program) (2013AA0400701), Shanghai Science and Technology Research Projects (12dz1125100), National Science and Technology Support Project during the 12th Five-Year Plan Period (2012BAF05B00), Shanghai Leading Academic Discipline Project (B504), Shanghai Natural Science Foundation (14ZR1421800), the State Key Laboratory of Synthetical Automation for Process Industries (PAL-N201404)

本文责任编辑 刘德荣

Recommended by Associate Editor LIU De-Rong

1. 华东理工大学化工过程先进控制和优化技术教育部重点实验室 上海 200237 2. 上海交通大学电工与电子技术中心 上海 200240

1. Key Laboratory of Advanced Control and Optimization for

在工程优化、计算机网络和人工智能等领域常常会遇到大规模、非线性、多极值等问题, 这类问题很难用传统数学手段处理. 智能优化算法依据计算机的迭代计算能力, 不依赖优化问题本身, 很好地解决了这一问题. 从 1975 年遗传算法 (Genetic algorithm, GA)^[1] 提出至今, 涌现出许多智能优化算法, 如粒子群算法 (Particle swarm optimization, PSO)^[2–5]、差分进化算法 (Differential evolution, DE)^[6–7]、群搜索算法 (Group search optimization, GSO)^[8] 和人工蜂群算法 (Artificial bee colony, ABC)^[9–11] 等. 然而这些算法在搜索和寻优性能方面还存在一些缺陷, 例如粒子群算法的求解是多样性逐渐丧失的过程, 所以粒子群算法局部寻优能力较差且易于发生早熟收敛现象. 因此, 研究者将一些新理论、新方法引入其中, 为智能算法的发

Chemical Processes, East China University of Science and Technology, Shanghai 200237 2. Center of Electrical and Electronic Technology, Shanghai Jiao Tong University, Shanghai 200240

展提供了新的思路.

教学优化算法 (TLBO) 是 Rao 等于 2011 年提出的新型优化算法^[12]. 该算法仅有群成员数和迭代代数两个参数, 需要设置的参数少, 避免了因参数设置不当造成的计算能力增加或者陷入局部最优的问题^[13-14]. TLBO 算法自提出以来, 引起了许多学者的关注. 2012 年 Rao 等提出了精英 TLBO 算法 (Elitist TLBO, ETLBO)^[15], ETLBO 算法在解决复杂约束问题上, 显示出了很好的性能. Rajasekhar 等提出了相对精英教学优化算法 (ETLOBA)^[16], 该算法在寻优精度及收敛速度上比 TLBO 算法有所改进. 然而, 上述算法的寻优精度及稳定性还不够高. 针对上述问题, 本文提出了反馈精英教学优化算法 (Feedback ETLBO, FETLBO). 在 ETLBO 算法的基础上, 在学生阶段之后加入了反馈阶段, 给学生提供了与教师交流的机会, 学生主动且有目的的与教师交流进行查缺补漏, 这样学生的学习方式在教师阶段的“教”过程和学生阶段的学生互相交流过程基础上有所增加, 学生的学习方式更加多样化, 提高算法的全局搜索能力. 同时, 反馈阶段是选举成绩较差的学生与教师交流, 成绩差的学生迅速向教师靠拢, 使算法后期进行局部精细搜索, 算法搜索更加细致, 提高了算法精度. 反馈阶段的加入不但增加了 ETLBO 算法的搜索细致性, 还提高了算法的全局搜索能力, 改善了该算法的寻优精度及稳定性. 为验证改进算法的性能, 本文选用 6 个无约束函数及 5 个约束函数进行测试, 结果表明 FETLBO 算法不但有更好的寻优精度和稳定性, 而且有更好的处理约束问题的能力. 最后将 FETLBO 算法应用于拉压弹簧优化设计问题和 0-1 背包问题, 也取得了较好的优化效果.

1 基于反馈精英教学优化算法 (FETLBO)

1.1 精英 TLBO 算法

为了理解 TLBO 算法, 假设有两个不同的教师 T_1 和 T_2 , 他们在不同的班级教授同等内容的科目, 两个班级学生的初始水平一样. 图 1 表示了两个班级学生分数在教师教学下的分布, 曲线 1 和曲线 2 分别是在教师 T_1 和 T_2 教学下的分数分布, M_1 和 M_2 分别是曲线 1 和曲线 2 的平均值. 假设分数服从正态分布式 (1).

从图 1 可以看出, 曲线 2 比曲线 1 有较高的平均值, 可以说教师 T_2 在教学上优于 T_1 . 除了教师的帮助外, 学生们还通过互相交流来提高分数. 图 2 中的曲线 A 表示了一个班级中的学生获得的分数分布模型. 教师是社会中最具有知识的人, 所以获得最高分数的学生作为教师, 如图 2 中的 T_A . 教师努力向

学生们传授知识, 这样就可以增加整个班级的平均分. 正如图 2 所示, 教师 T_A 通过教授学生们知识努力将班级的平均分由 M_A 靠近新的平均分 M_B , 这样学生们又需要比他们知识多的新教师, 如图 2 中的新曲线 B 上的新教师 T_B .

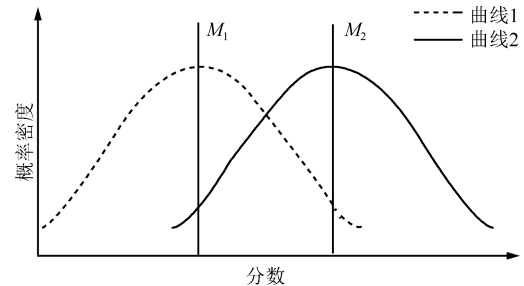


图 1 不同教师教学下的学生分数分布

Fig. 1 Distribution curves of learners' marks taught by two different teachers

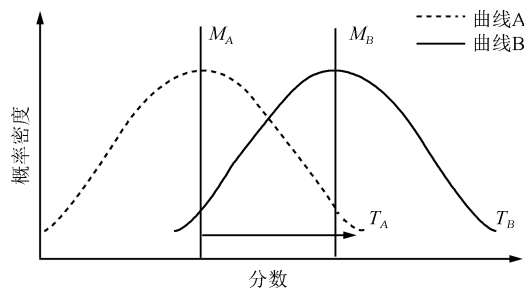


图 2 学生获得分数的分布曲线

Fig. 2 Distribution curve of marks obtained for a group of learners

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \quad (1)$$

其中, σ^2 是方差, μ 为均值, x 是正态分布的随机变量.

如其他受自然现象启发的算法一样, TLBO 算法也是一种基于种群的算法. 在 TLBO 算法中, 学生人数为算法的种群数, 学生学习的不同科目为自变量, 学生的学习结果为适应度值, 教师为当前最好解. TLBO 算法分为两阶段: 教师阶段和学生阶段, 教师阶段为学生从教师处学习知识, 学生阶段为学生之间互相交流学习知识. 教师阶段的输出作为学生阶段的输入.

1.1.1 教师阶段

在任意迭代次数 i , M_i 为平均值, T_i 为教师, 教师 T_i 尽力使平均值 M_i 接近自己的水平, 所以新的平均值 M_{new} 接近 T_i , 当前平均值与新的平均值之差由式 (2) 给出.

$$\text{Difference_Mean}_i = r_i(M_{\text{new}} - T_i M_i) \quad (2)$$

其中, r_i 为 0-1 的随机数, T_F 为教学因子, 它决定平均值改变的程度, T_F 由式 (3) 随机确定为 1 或 2.

$$T_F = \text{round}[1 + \text{rand}(0, 1)] \quad (3)$$

教师阶段根据式 (4) 来对当前解进行更新:

$$x_{\text{new},i} = x_{\text{old},i} + \text{Difference_Mean}_i \quad (4)$$

如果 x_{new} 优于 x_{old} , 则接受 x_{new} .

1.1.2 学生阶段

学生阶段为学生之间随机进行互相交流, 学生能够从比自己知识多的学生那获取新的知识. x 为优化问题自变量, $f(x)$ 为优化问题目标函数. 学生阶段过程如下: 教师阶段之后, 随机选择两个学生 x_i 和 x_h , 其中 $i \neq h$. 首先比较两个学生对应的目标函数值, 若 $f(x_i) < f(x_h)$, 则表示学生 x_i 优于学生 x_h , 那么 x_{new} 向 x_i 靠近, 如式 (5) 所示. 反之, 学生 x_h 优于学生 x_i , 则 x_{new} 向 x_h 靠近, 如式 (6) 所示.

$$x_{\text{new},i} = x_{\text{old},i} + \text{rand}_i(x_i - x_h), \quad f(x_i) < f(x_h) \quad (5)$$

$$x_{\text{new},i} = x_{\text{old},i} + \text{rand}_i(x_h - x_i), \quad f(x_h) < f(x_i) \quad (6)$$

在学生阶段过程之后, 比较新解 x_{new} 与当前解 x_{old} , 如果 x_{new} 优于 x_{old} , 则接受 x_{new} .

1.1.3 精英策略

精英策略广泛应用于群智能算法中来保留各代的最优解, 进而提高算法的收敛速度. 精英 TLBO 算法通过引入精英策略, 保留每代中的最优解, 由精英个体取代较差个体, 加快了算法的收敛速度. 为避免算法陷入局部最优解, 精英 TLBO 算法在进行下一次迭代前随机对精英个体进行变异操作, 那么在迭代后期, 较好的个体得到保留, 每代结束后的最优解作为教师进行下一次迭代.

1.2 反馈精英 TLBO 算法

在精英 TLBO 算法中, 学生仅通过教师教学或者与学生们交流来提高分数, 学习方式比较单一. 然而在实际的学生学习过程中, 学生往往也与教师主动且有目的的进行反馈交流, 通过反馈针对自己所学知识进行查缺补漏能够得到更多的知识, 这样便能进一步提高学生分数. 因此, 本文在精英 TLBO 算法基础上, 引入反馈阶段来提高算法的寻优精度及稳定性.

反馈阶段加入在学生阶段之后, 这样学生提高分数除了通过教师教学和学生之间互相交流外, 还有学生直接与教师进行反馈交流. 反馈阶段过程如

下: 在学生阶段之后, 随机选择两个学生 x_i 和 x_d , 其中 $i \neq d$. 比较两个学生对应的目标函数值, 若 $f(x_i) < f(x_d)$, 则表示学生 x_i 优于 x_d , 那么选取学生 x_d 与教师进行反馈交流, 如式 (7) 所示. 反之, 选取学生 x_i 与教师进行反馈交流, 如式 (8) 所示.

$$x_{\text{new},i} = x_{\text{old},i} + \text{rand}_i(M_{\text{new}} - x_d), \quad f(x_i) < f(x_d) \quad (7)$$

$$x_{\text{new},i} = x_{\text{old},i} + \text{rand}_i(M_{\text{new}} - x_i), \quad f(x_d) < f(x_i) \quad (8)$$

反馈阶段过程之后, 比较新解 x_{new} 与当前解 x_{old} , 如果 x_{new} 优于 x_{old} , 则接受 x_{new} .

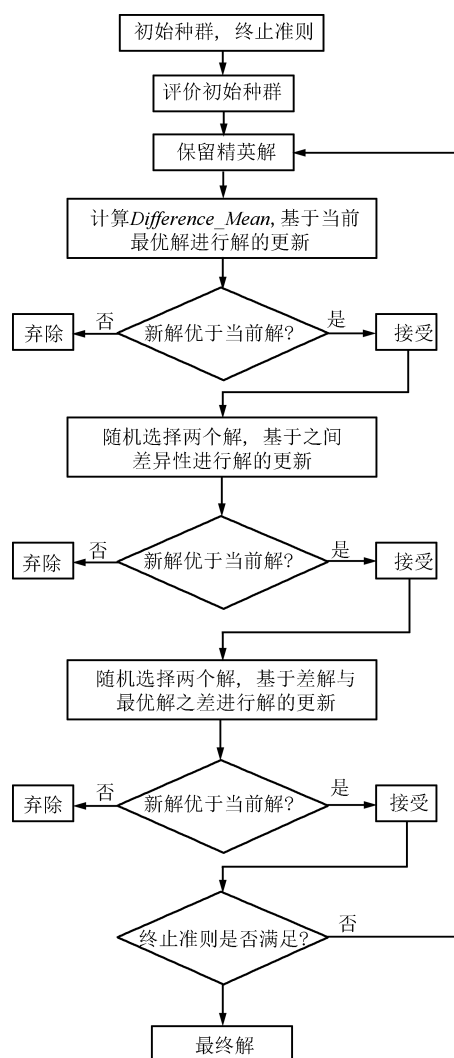


图3 FETLBO 算法流程图

Fig. 3 Flow chart of FETLBO

上述反馈过程的加入, 增加了学生的学习方式, 保证学生的多样性, 提高算法的全局搜索性能. 同时反馈阶段使较差学生快速向当前最优个体教师靠拢, 搜索范围快速收率到最优解附近, 在算法终止条件

迭代代数一定时, 算法在后期进行局部精细搜索的代数相对增加, 这样算法的寻优精度和稳定性都会有所提高.

反馈精英教学算法 (FETLBO) 用于优化问题的步骤总结如下, 其流程如图 3 所示.

步骤 1. 定义优化问题和初始化优化问题参数. 初始化群成员数、迭代代数、自变量个数以及优化问题的限制条件.

步骤 2. 初始种群. 根据群成员数及自变量个数随机产生初始种群.

步骤 3. 评价种群并保留精英解.

步骤 4. 教师阶段, 根据式 (4) 进行教师阶段的教学过程.

步骤 5. 学生阶段, 根据式 (5) 和 (6) 学生之间随机进行互相交流, 提高成绩.

步骤 6. 反馈阶段, 根据式 (7) 和 (8) 学生与老师进行反馈交流, 提高学生成绩.

步骤 7. 精英解取代较差解.

步骤 8. 随机对精英解进行变异操作.

步骤 9. 重复步骤 3~8, 直到满足结束条件为止.

2 数值仿真及分析

2.1 无约束测试函数仿真及分析

选用 6 个常用的无约束测试函数, 分别在 30 和 100 维情况下, 将 ETLBO 算法、粒子群算法 (PSO)、差分进化算法 (DE) 及文献 [17] 提出的差分进化群搜索混合算法 (DEGSO) 进行比较. PSO 算法中, 惯性权重为 0.5, $c_1 = c_2 = 1.49$. DE 算法

中, 变异因子 0.5, 交叉概率 0.1. DEGSO 算法中, 成员初始角 $\pi/4$, 常数 $a = \sqrt{n+1}$ 取整, 最大搜索角度 $\theta_{\max} = \pi/a^2$, 最大的搜索转角 $\alpha_{\max} = \pi/2a^2$, 上述三种算法种群数均为 50, ETLBO 和 FETLBO 种群数为 10. 上述各算法迭代次数均为 1000 代.

采用表 1 中的 6 个常用无约束测试函数分别在 30、100 维的情况下对各个算法进行 30 次测试, 得到的测试结果如表 2 所示. 从表 2 可以看出, 在 30 和 100 维的情况下, FETLBO 算法在平均值和标准差方面都要好于其他算法, 对于 f_3 和 f_5 , ETLBO 和 FETLBO 均能达到全局最优. 对于 f_2 , FETLBO 算法的标准差远小于 ETLBO 的标准差. 可见, 对于无约束问题无论是在低维还是高维情况下, FETLBO 算法在寻优精度及稳定性方面都要优于 ETLBO 和其他算法.

表 1 无约束测试函数

Table 1 Unconstrained benchmark functions

函数	变量范围	最优值
$f_1(x) = \sum_{i=1}^D x_i^2$	$(-100, 100)$	0
$f_2(x) = -20 \exp(-0.2 \sqrt{\frac{1}{D} \sum_{i=1}^D x_i^2}) - \exp(\sum_{i=1}^D \cos(2\pi x_i)/D) + 20 + e$	$(-32, 32)$	0
$f_3(x) = \frac{1}{4000} \sum_{i=1}^D x_i^2 - \prod_{i=1}^D \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	$(-600, 600)$	0
$f_4(x) = \sum_{i=1}^{D-1} [100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2]$	$(-10, 10)$	0
$f_5(x) = \sum_{i=1}^D (\sum_{j=1}^i x_j^2)^2$	$(-100, 100)$	0
$f_6(x) = \sum_{i=1}^D x_i + \prod_{i=1}^D x_i $	$(-10, 10)$	0

表 2 无约束测试函数对比结果平均值 (标准差)

Table 2 Comparison of results for unconstrained benchmark functions mean (std)

函数	维数	PSO	DE	DEGSO	ETLBO	FETLBO
f_1	30	4.06E-015(1.25E-014)	1.82E-013(6.54E-014)	2.70E-010(3.72E-010)	3.97E-165(2.54E-164)	3.43E-231(3.21E-231)
	100	4.78E+002(7.47E+002)	1.35E-012(4.89E-013)	1.05E-004(3.32E-004)	1.50E-163(2.26E-163)	2.61E-230(1.12E-230)
f_2	30	2.23E+000(5.83E+000)	3.14E-007(6.01E-008)	2.22E-006(1.04E-006)	6.21E-015(1.87E-015)	4.44E-015(0.00E+000)
	100	3.10E+000(5.78E+000)	8.47E-007(1.51E-007)	1.73E-005(2.98E-005)	6.21E-015(1.99E-014)	4.44E-015(0.00E+000)
f_3	30	1.56E-002(3.98E-002)	1.84E-013(3.15E-013)	2.59E-008(6.98E-008)	0.00E+000(0.00E+000)	0.00E+000(0.00E+000)
	100	1.43E-002(1.32E-002)	3.36E-012(7.41E-012)	3.81E-006(2.53E-006)	0.00E+000(0.00E+000)	0.00E+000(0.00E+000)
f_4	30	3.45E+001(2.34E+001)	4.09E+001(1.87E+001)	3.53E+001(1.13E+001)	2.67E+001(2.26E-001)	2.57E+001(1.23E-001)
	100	3.35E+001(2.58E+001)	4.70E+001(1.76E+001)	5.42E+001(3.15E+001)	2.67E+001(3.11E-001)	2.60E+001(2.16E-001)
f_5	30	2.41E-027(1.04E-026)	3.82E-023(3.06E-023)	9.11E-020(2.83E-019)	0.00E+000(0.00E+000)	0.00E+000(0.00E+000)
	100	2.83E-025(1.51E-024)	2.21E-021(1.02E-021)	3.55E-010(1.12E-009)	0.00E+000(0.00E+000)	0.00E+000(0.00E+000)
f_6	30	6.89E-006(1.70E-005)	1.84E-007(5.44E-008)	4.89E-007(6.16E-007)	1.26E-083(1.76E-083)	4.16E-116(1.07E-115)
	100	1.39E-005(3.99E-005)	6.27E-007(1.10E-007)	7.83E-007(2.23E-007)	2.73E-083(3.96E-083)	4.49E-115(8.74E-115)

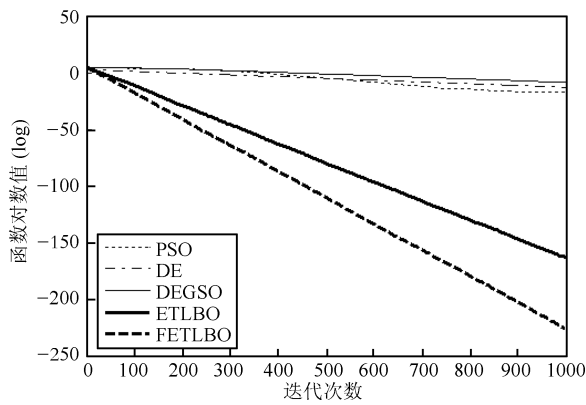


图 4 各算法在 f_1 中的表现

Fig. 4 Performance of algorithms in f_1

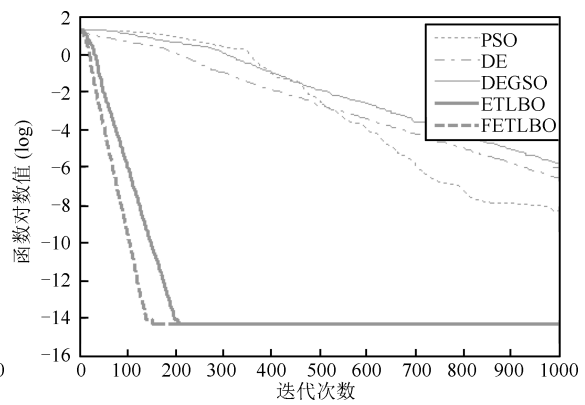


图 5 各算法在 f_2 中的表现

Fig. 5 Performance of algorithms in f_2

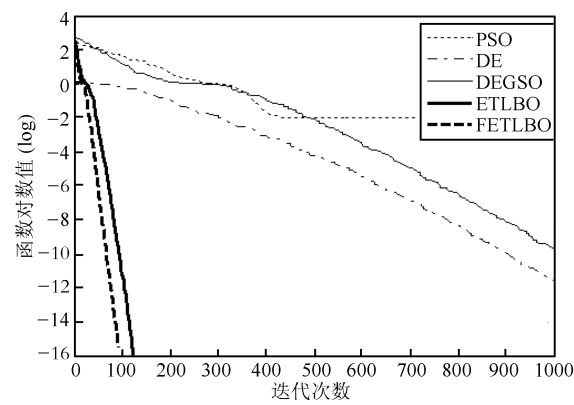


图 6 各算法在 f_3 中的表现

Fig. 6 Performance of algorithms in f_3

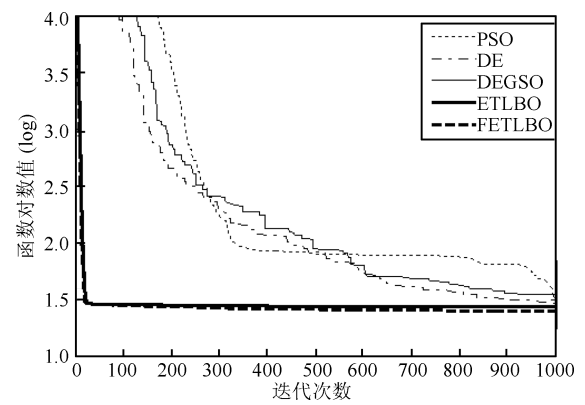


图 7 各算法在 f_4 中的表现

Fig. 7 Performance of algorithms in f_4

图 4~7 为各算法分别在 $f_1 \sim f_4$ 中 30 维情况下的一次寻优曲线, 由图可以看出, FETLBO 算法在寻优精度方面好于其他算法, 达到了改进的目的.

2.2 约束测试函数仿真及分析

选用 5 个带约束的测试函数 ($f_7 \sim f_{11}$) 进行求解, 测试函数表达式及变量范围见附录, 参数设置按照文献 [15] 中进行设置: 种群数 25, 迭代次数为 4700 代.

表 3 列出了 FETLBO 与 ETLBO 算法在解决有约束函数连续运行 30 次得到的结果. 其中 ETLBO 算法的结果来自文献 [15]. 从结果可以看出, 对于 f_8 和 f_{11} , 本论文提出的 FETLBO 算法比 ETLBO 算法能得到更好的解并且稳定性更好. 对于 f_{10} , FETLBO 算法获得解的精度优于 ETLBO 算法. 对于 f_7 和 f_9 , 两种算法都能稳定地达到全局最优解. 由此可见, 反馈阶段的引入加强了算法的全局搜索能力和后期的细微搜索, 提高了寻优精度和稳定性.

表 3 约束函数测试结果

Table 3 Results of constrained benchmark functions

函数 (全局最优)	算法	最优值	平均值	标准差
f_7 (-30 665.539)	ETLBO	-30 665.539	-30 665.539	0
	FETLBO	-30 665.539	-30 665.539	0
f_8 (7 049.248)	ETLBO	7 218.258	7370.191	125
	FETLBO	7 060	7 220	110
f_9 (-6 961.814)	ETLBO	-6 961.814	-6 961.814	0
	FETLBO	-6 961.814	-6 961.814	0
f_{10} (-15)	ETLBO	-15	-13.35	1.58
	FETLBO	-15	-13.392	1.74
f_{11} (24.3062)	ETLBO	24.318	24.4926	0.2451
	FETLBO	24.308	24.3515	0.1993

3 FETLBO 算法应用研究

3.1 优化拉压弹簧设计

拉压弹簧优化设计问题的目标是在满足最小挠度、剪应力和振动频率等约束下最小化其质量, 设计变量分别为弹簧线圈直径 $d(x_1)$ 、簧圈平均直径 $D(x_2)$ 和绕线圈数 $P(x_3)$, 如图 8 所示.

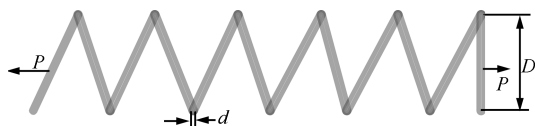


图 8 拉压弹簧简图

Fig. 8 Sketch of tension/compression string

问题的目标函数和约束条件为:

$$\begin{aligned} \min f(x) &= (x_3 + 2)x_1^2 x_2 & (9) \\ \text{s.t.} \quad & \begin{cases} g_1(x) = 1 - \frac{x_2^3 x_3}{71785 x_1^4} \leq 0 \\ g_2(x) = \frac{4x_2^2 - x_1 x_2}{12566(x_1^3 x_2 - x_1^4)} + \frac{1}{5108 x_1^2} - 1 \leq 0 \\ g_3(x) = 1 - \frac{140.45 x_1}{x_2^2 x_3} \leq 0 \\ g_4(x) = \frac{x_1 + x_2}{1.5} - 1 \leq 0 \\ 0.05 \leq x_1 \leq 2, 0.25 \leq x_2 \leq 1.3, \\ 2 \leq x_3 \leq 15 \end{cases} & (10) \end{aligned}$$

利用 FETLBO 算法对拉压弹簧优化设计问题进行求解, 并与协同进化粒子群算法 (Co-evolutionary particle swarm optimization, CPSO)^[18], 社会和文化算法 (Society and civilization algorithm, SCA)^[19], 加速自适应权重模型 (Accelerating adaptive trade-off model, AATM)^[20], 协同进化差分进化算法 (Co-evolutionary differential evolution, CDE)^[21], ETLBO 算法比较结果如表 4 所示, 对比可以看出 FETLBO 算法的优化结果要好于其他算法。

表 4 拉压弹簧优化设计问题结果

Table 4 Results of tension/compression string design

算法	$x_1(d)$	$x_2(D)$	$x_3(P)$	$f(x)$
CPSO	0.051728	0.357644	11.244543	0.0126747
SCA	0.052160	0.368158	10.648442	0.0126692
AATM	0.051813	0.359690	11.119252	0.0126682
CDE	0.051609	0.354714	11.410831	0.0126702
ETLBO	0.051565	0.353713	11.468954	0.01266738
FETLBO	0.051691	0.356758	11.286578	0.01266523

3.2 0-1 背包问题

0-1 背包问题是一类在给定约束条件的情况下, 求最大值的组合优化非确定多项式 (NP) 完全难题。背包问题可描述如下: 已知 N 个物体, 第 $j(j = 1, 2, \dots, N)$ 个物体的重量和价值分别为 w_j 和 c_j , 背包所能容纳的物体总重量上限为 v , 问题的目标就是要在不超过背包最大承重量的前提下, 使

背包中的物体价值总和最大化。具体数学模型如下:

$$\max f(x) = \sum_{j=1}^N c_j x_j \quad (11)$$

$$\text{s.t.} \quad \sum_{j=1}^N w_j x_j \leq v, x_j = 0, 1, j = 1, 2, \dots, N \quad (12)$$

以文献 [22] 的 3 个低维 0-1 背包问题为研究对象, 问题数据如下 $f_{12} \sim f_{14}$:

f_{12} : $w = (6, 5, 9, 7)$, $c = (9, 11, 13, 15)$, $v = 20$

f_{13} : $w = (2, 4, 6, 7)$, $c = (6, 10, 12, 13)$, $v = 11$

f_{14} : $w = (31, 10, 20, 19, 4, 3, 6)$, $c = (70, 20, 39, 37, 7, 5, 10)$, $v = 50$

其中, w 为物品重量, c 为物品价值, v 为背包最大承重量。

FETLBO 算法、ETLBO 算法参数设置为种群数 20, 迭代次数 1000 代。连续运行 50 次, 将结果与文献 [23] 提出的改进的和谐搜索算法 (Improved harmony search, IHS) 结果比较见表 5 所示 (IHS 的结果取自文献 [22])。从表 5 可以看出, 三种算法均能找到问题的全局最优解, 但 FETLBO 算法的稳定性及精度更好, 尤其问题 f_{12} 和 f_{13} 均能稳定地找到全局最优解。

表 5 低维背包问题结果比较

Table 5 Comparison of results for small scale KP

函数	算法	最优值	平均值	标准差
f_{12}	IHS	35	34.58	1.68
	ETLBO	35	34.93	0.365
	FETLBO	35	35	0
f_{13}	IHS	23	23	0
	ETLBO	23	23	0
	FETLBO	23	23	0
f_{14}	IHS	107	103.9	4.48
	ETLBO	107	105.2	3.353
	FETLBO	107	105.9	1.388

表 6 50 维背包问题结果比较

Table 6 Comparison of results for 50 scale KP

算法	最优值 (总价值/总重量)
GA	3077/999
PSO	2825/1000
ETLBO	3092/1000
FETLBO	3094/999

为进一步说明 FETLBO 算法的可行性, 选取文献 [24] 中给定的 1 个经典 50 维的背包实例进行求解 (数据见文献 [24] 仿真实例 2)。FETLBO 和 ETLBO 算法求得的最好结果与文献 [24] 报道的

GA 和 PSO 的结果比较如表 6 所示.

由表 6 可知, FETLBO、ETLBO 算法结果要优于遗传算法和标准粒子群算法, 而 FETLBO 算法结果优于 ETLBO 算法, 说明 FETLBD 算法能够很好地应用在 0-1 背包问题上.

4 结论

本文针对精英 TLBO 算法寻优精度低及稳定性差的问题提出了反馈精英教学算法 (FETLBO), 该算法通过引入反馈阶段增加了算法的搜索遍历性, 弥补了精英 TLBO 算法寻优精度低及稳定性差的缺陷. 通过与 ETLBO 算法和其他算法在 6 个无约束测试函数和 5 个约束测试函数中的比较, 结果表明该算法具有良好的寻优精度和稳定性. 最后将 FETLBO 算法应用于拉压弹簧设计问题和 0-1 背包问题, 结果表明 FETLBO 算法比其他算法具有更好的优化效果.

附录

约束测试函数:

$$\min f_7(x) = 5.3578547x_3^2 + 0.8356891x_1x_5 + 37.293239x_1 - 40792.141 \quad (\text{A1})$$

$$\left\{ \begin{array}{l} g_1(x) = 85.334407 + 0.0056858x_2x_5 + 0.0006262x_1x_4 - 0.0022053x_3x_5 - 92 \leq 0 \\ g_2(x) = -85.334407 - 0.0056858x_2x_5 - 0.0006262x_1x_4 + 0.0022053x_3x_5 \leq 0 \\ g_3(x) = 80.51249 + 0.007137x_2x_5 + 0.0029955x_1x_2 + 0.0021813x_3^2 - 110 \leq 0 \\ g_4(x) = -80.51249 - 0.007137x_2x_5 - 0.0029955x_1x_2 - 0.0021813x_3^2 + 90 \leq 0 \\ g_5(x) = 9.300961 + 0.0047026x_3x_5 + 0.0012547x_1x_3 + 0.0019085x_3x_4 - 25 \leq 0 \\ g_6(x) = -9.300961 - 0.0047026x_3x_5 - 0.0012547x_1x_3 - 0.0019085x_3x_4 + 20 \leq 0 \end{array} \right. \quad (\text{A2})$$

其中, $l_i \leq x_i \leq u_i$, $i = 1, \dots, 5$, $l = (78, 33, 27, 27, 27)$, $u = (102, 45, 45, 45, 45)$.

$$\min f_8(x) = x_1 + x_2 + x_3 \quad (\text{A3})$$

$$\left\{ \begin{array}{l} g_1(x) = -1 + 0.0025(x_4 + x_6) \leq 0 \\ g_2(x) = -1 + 0.0025(x_5 + x_7 - x_4) \leq 0 \\ g_3(x) = -1 + 0.01(x_8 - x_5) \leq 0 \\ g_4(x) = -x_1x_6 + 833.33252x_4 + 100x_1 - 83333.333 \leq 0 \\ g_5(x) = -x_2x_7 + 1250x_5 + x_2x_4 - 1250x_4 \leq 0 \\ g_6(x) = -x_3x_8 + 1250000 + x_3x_5 - 2500x_5 \leq 0 \end{array} \right. \quad (\text{A4})$$

其中, $l_i \leq x_i \leq u_i$, $i = 1, \dots, 8$, $l = (100, 1000, 1000, 10, 10, 10, 10, 10)$, $u = (10000, 10000, 10000, 1000, 1000, 1000, 1000, 1000)$.

$$\min f_9(x) = (x_1 - 10)^3 + (x_2 - 20)^3 \quad (\text{A5})$$

$$\left\{ \begin{array}{l} g_1(x) = -(x_1 - 5)^2 - (x_2 - 5)^2 + 100 \leq 0 \\ g_2(x) = (x_1 - 6)^2 + (x_2 - 5)^2 - 82.81 \leq 0 \end{array} \right. \quad (\text{A6})$$

其中, $13 \leq x_1 \leq 100$, $0 \leq x_2 \leq 100$.

$$\min f_{10}(x) = 5 \sum_{i=1}^4 x_i - 5 \sum_{i=1}^4 x_i^2 - \sum_{i=5}^{13} x_i \quad (\text{A7})$$

$$\left\{ \begin{array}{l} g_1(x) = 2x_1 + 2x_2 + x_{10} + x_{11} - 10 \leq 0 \\ g_2(x) = 2x_1 + 2x_3 + x_{10} + x_{12} - 10 \leq 0 \\ g_3(x) = 2x_2 + 2x_3 + x_{11} + x_{12} - 10 \leq 0 \\ g_4(x) = -8x_1 + x_{10} \leq 0 \\ g_5(x) = -8x_2 + x_{11} \leq 0 \\ g_6(x) = -8x_3 + x_{12} \leq 0 \\ g_7(x) = -2x_4 - x_5 + x_{10} \leq 0 \\ g_8(x) = -2x_6 - x_7 + x_{11} \leq 0 \\ g_9(x) = -2x_8 - x_9 + x_{12} \leq 0 \end{array} \right. \quad (\text{A8})$$

其中, $0 \leq x_i \leq 1$ ($i = 1, \dots, 9$), $0 \leq x_i \leq 100$ ($i = 10, 11, 12$), $0 \leq x_{13} \leq 1$.

$$\min f_{11}(x) = x_1^2 + x_2^2 + x_1x_2 - 14x_1 - 16x_2 + (x_3 - 10)^2 + 4(x_4 - 5)^2 + (x_5 - 3)^2 + 2(x_6 - 1)^2 + 5x_7^2 + 7(x_8 - 11)^2 + 2(x_9 - 10)^2 + (x_{10} - 7)^2 + 45 \quad (\text{A9})$$

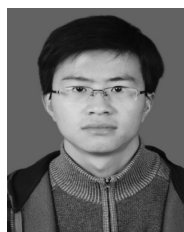
$$\left\{ \begin{array}{l} g_1(x) = -105 + 4x_1 + 5x_2 - 3x_7 + 9x_8 \leq 0 \\ g_2(x) = 10x_1 - 8x_2 - 17x_7 + 2x_8 \leq 0 \\ g_3(x) = -8x_1 + 2x_2 + 5x_9 - 2x_{10} - 12 \leq 0 \\ g_4(x) = 3(x_1 - 2)^2 + 4(x_2 - 3)^2 + 2x_3^2 - 7x_4 - 120 \leq 0 \\ g_5(x) = 5x_1^2 + 8x_2 + (x_3 - 6)^2 - 2x_4 - 40 \leq 0 \\ g_6(x) = x_1^2 + 2(x_2 - 2)^2 - 2x_1x_2 + 14x_5 - 6x_6 \leq 0 \\ g_7(x) = 0.5(x_1 - 8)^2 + 2(x_2 - 4)^2 + 3x_5 - x_6 - 30 \leq 0 \\ g_8(x) = -3x_1 + 6x_2 + 12(x_9 - 8)^2 - 7x_{10} \leq 0 \end{array} \right. \quad (\text{A10})$$

其中, $-10 \leq x_i \leq 10$ ($i = 1, \dots, 10$).

References

- 1 Holland J H. *Adaptation in Natural and Artificial Systems*. Ann Arbor: University of Michigan Press, 1975. 1-53
- 2 Kennedy J, Eberhart R C. Particle swarm optimization. In: *Proceedings of IEEE International Conference on Neural Networks*. Perth, Australia: IEEE, 1995. 1942-1948

- 3 Pan Feng, Chen Jie, Xin Bin, Zhang Juan. Several characteristics analysis of particle swarm optimizer. *Acta Automatica Sinica*, 2009, **35**(7): 1010–1016
(潘峰, 陈杰, 辛斌, 张娟. 粒子群优化方法若干特性分析. 自动化学报, 2009, **35**(7): 1010–1016)
- 4 Pan Feng, Chen Jie, Gan Ming-Gang, Cai Tao, Tu Xu-Yan. Model analysis of particle swarm optimizer. *Acta Automatica Sinica*, 2006, **32**(3): 368–377
(潘峰, 陈杰, 甘明刚, 蔡涛, 涂序彦. 粒子群优化算法模型分析. 自动化学报, 2006, **32**(3): 368–377)
- 5 Jin Xin-Lei, Ma Long-Hua, Wu Tie-Jun, Qian Ji-Xin. Convergence analysis of the particle swarm optimization based on stochastic processes. *Acta Automatica Sinica*, 2007, **33**(12): 1263–1268
(金欣磊, 马龙华, 吴铁军, 钱积新. 基于随机过程的 PSO 收敛性分析. 自动化学报, 2007, **33**(12): 1263–1268)
- 6 Qian W Y, Li A J. Adaptive differential evolution algorithm for multi-objective optimization problems. *Applied Mathematics and Computation*, 2008, **201**(1–2): 431–440
- 7 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
- 8 He S, Wu Q H, Saunders J R. Group search optimizer: an optimization algorithm inspired by animal searching behavior. *IEEE Transactions on Evolutionary Computation*, 2009, **13**(5): 973–990
- 9 Karaboga D, Basturk B. On the performance of artificial bee colony (ABC) algorithm. *Applied Soft Computing*, 2008, **8**(1): 687–697
- 10 Karaboga D, Basturk B. A comparative study of artificial bee colony algorithm. *Applied Mathematics and Computation*, 2009, **214**(1): 108–132
- 11 Karaboga D, Basturk B. A Powerful and efficient algorithm for numerical function optimization: artificial bee colony algorithm. *Journal of Global Optimization*, 2007, **39**(3): 459–471
- 12 Rao R V, Savsani V J, Vakharia D P. Teaching-learning-based optimization: a novel method for constrained mechanical design optimization problems. *Computer Aided Design*, 2011, **43**(3): 303–315
- 13 Rao R V, Savsani V J, Vakharia D P. Teaching-learning-based optimization: an optimization method for continuous non-linear large scale problems. *Information Sciences*, 2012, **183**(1): 1–15
- 14 Niknam T, Azizipanah-Abarghooee R, Narimani M R. A new multi objective optimization approach based on TLBO for location of automatic voltage regulators in distribution systems. *Engineering Applications of Artificial Intelligence*, 2012, **25**(8): 1577–1588
- 15 Rao R V, Patel V. An elitist teaching-learning-based optimization algorithm for solving complex constrained optimization problems. *International Journal of Industrial Engineering Computations*, 2012, **3**(4): 535–560
- 16 Rajasekhar A, Rani R, Ramya K, Abraham A. Elitist teaching-learning opposition based algorithm for global optimization. In: Proceedings of IEEE International Conference on Systems, Man, and Cybernetics. Seoul, Korea: IEEE, 2012. 1124–1129
- 17 Nian Xiao-Yu, Wang Zhen-Lei, Qian Feng. A hybrid algorithm based on differential evolution and group search optimization and its application on ethylene cracking furnace. *Chinese Journal of Chemical Engineering*, 2013, **21**(5): 537–543
- 18 He Q, Wang L. An effective co-evolutionary particle swarm optimization for constrained engineering design problems. *Engineering Applications of Artificial Intelligence*, 2007, **20**(1): 89–99
- 19 Ray T, Liew K M. Society and civilization: an optimization algorithm based on the simulation of social behavior. *IEEE Transactions on Evolutionary Computation*, 2003, **7**(4): 386–396
- 20 Wang Y, Cai Z X, Zhou Y R. Accelerating adaptive trade-off model using shrinking space technique for constrained evolutionary optimization. *International Journal for Numerical Methods in Engineering*, 2009, **77**(11): 1501–1534
- 21 Huang F Z, Wang L, He Q. An effective co-evolutionary differential evolution for constrained optimization. *Applied Mathematics and computation*, 2007, **186**(1): 340–356
- 22 Zou D X, Gao L Q, Li S, Wu J H. Solving 0-1 knapsack problem by a novel global harmony search algorithm. *Applied Soft Computing*, 2011, **11**(2): 1556–1554
- 23 Mahdavi M, Fesanghary M, Damangir E. An improved harmony search algorithm for solving optimization problems. *Applied Mathematics and Computation*, 2007, **188**(2): 1567–1579
- 24 Gao Fang, Cui Gang, Wu Zhi-Bo, Liu Hong-Wei, Yang Xiao-Zong. Virus-evolutionary particle swarm optimization algorithm for knapsack problem. *Journal of Harbin Institute of Technology*, 2009, **41**(6): 103–107
(高芳, 崔刚, 吴智博, 刘宏伟, 杨孝宗. 求解背包问题的病毒协同进化粒子群算法. 哈尔滨工业大学学报, 2009, **41**(6): 103–107)



于坤杰 华东理工大学信息科学与工程学院博士研究生。主要研究方向为智能优化算法。

E-mail: yukunjie1990@gmail.com

(YU Kun-Jie Ph.D. candidate at the School of Information Science and Engineering, East China University of Science and Technology. His research interest is intelligent optimization algorithms.)



王昕 上海交通大学副教授。主要研究方向为多模型自适应控制, 智能解耦和复杂工业过程建模、控制与优化。

E-mail: wangxin26@sjtu.edu.cn

(WANG Xin Associate professor at Shanghai Jiao Tong University. His research interest covers multiple models adaptive control, multivariable intelligent decoupling control, and modeling, control and optimization of complex industrial processes.)



王振雷 华东理工大学教授。主要研究方向为智能控制, 复杂系统的建模及特征分析, 故障诊断和智能优化算法。本文通信作者。

E-mail: wangzhen.l@ecust.edu.cn

(WANG Zhen-Lei Professor at East China University of Science and Technology. His research interest covers intelligent control, modeling and analysis the characteristics of complex systems, fault diagnosis, and intelligent optimization algorithms research. Corresponding author of this paper.)