

基于非线性收敛因子和标杆管理的 改进教与学优化算法

陈雪芬, 叶春明

(上海理工大学 管理学院, 上海 200093)

摘要: 针对教与学优化算法寻优精度低、收敛速度慢以及局部最优避免性弱的问题, 提出了一种改进教与学优化算法 (MTLBO)。在算法的教学和学习阶段, 分别引入了非线性收敛因子调整策略和标杆管理策略。基于 2 种策略的随机组合形成了 3 种不同的 MTLBOs, 与标准教与学优化算法 (TLBO) 的对比实验结果表明, 3 种 MTLBO 均优于 TLBO, 其中, 引入 2 种策略的 MTLBO3 取得了最佳的数值实验结果, 其远优于原始 TLBO。为进一步验证提出算法的有效性, 与其他著名的群智能优化算法进行了数值实验对比。数值结果和收敛曲线表明, MTLBO3 的寻优性能明显优于其他对比方法, 具有更高的求解精度、更快的收敛速度以及更佳的局部最优避免能力。最后, 使用有约束的工程优化问题进一步验证了提出算法的有效性。

关键词: 群智能优化算法; 教与学优化算法; 收敛因子; 标杆管理

中图分类号: TP 301.6

文献标志码: A

Modified teaching-learning-based optimization algorithm based on the nonlinear convergence factor and benchmarking management

CHEN Xuefen, YE Chunming

(Business School, University of Shanghai for Science and Technology, Shanghai 200093, China)

Abstract: A modified teaching-learning-based optimization algorithm (MTLBO) was proposed to solve the shortcomings of standard teaching-learning-based optimization algorithm (TLBO), such as low optimization accuracy, slow convergence speed and weak avoidance of local optimization. In the teaching and learning stages of the TLBO, the nonlinear convergence factor adjustment and benchmarking management strategies were introduced respectively. Based on the random combination of the two strategies, three different MTLBOs were constructed. Subsequently, the experimental results show that the three MTLBOs are better than the TLBO. Among them, the MTLBO3 with the two modified strategies achieves the best numerical results, which is much better than the original TLBO. In order to further verify the effectiveness of the proposed algorithm, numerical experiments are carried out

收稿日期: 2021-10-03

基金项目: 国家自然科学基金资助项目 (71840003); 上海市哲学社会科学一般项目 (2022BGL10)

第一作者: 陈雪芬 (1997-), 女, 硕士研究生。研究方向: 工业工程。E-mail: 15902171827@163.com

通信作者: 叶春明 (1964-), 男, 教授。研究方向: 管理科学与工程、工业工程。E-mail: Yechm6464@163.com

with other well-known swarm intelligence optimization algorithms. Numerical results and convergence curves show that the optimization performance of MTLBO3 is significantly better than other comparison methods, with higher solution accuracy, faster convergence speed and better local optimization avoidance ability. Finally, the effectiveness of the proposed algorithm is further verified using constrained engineering optimization problems.

Keywords: *swarm intelligence optimization algorithm; teaching-learning-based optimization algorithm; convergence factor; benchmarking management*

群智能优化算法是一类基于种群的全局优化技术,同传统的精确算法不同,它们更擅长于求解具有大规模、非线性、多极值、不可导及不可微等特征的非常规问题。随着人工智能、大数据以及云计算等技术的快速发展,复杂优化问题的数量和规模与日俱增,这些问题往往难以用常规数学方法求解,而群智能算法则成为相对可行的解决方案。自1992年遗传算法(genetic algorithm, GA)^[1]提出以来,众多模拟自然规模或者物种动态演化的群智能算法相继被提出。在群智能算法“家族”中,具有很多知名的成员,它们深受研究者的青睐,如粒子群算法(particle swarm optimization, PSO)^[2]、和声搜索算法(harmony search algorithm, HS)^[3]、人工蜂群算法(artificial bee colony algorithm, ABC)^[4]、引力搜索算法(gravity search algorithm, GSA)^[5]、蛾焰优化算法(moth-flame optimization algorithm, MFO)^[6]、哈里斯鹰算法(Harris hawks optimization, HHO)^[7]和灰狼算法(grey wolf optimization algorithm)^[8]等。这些算法各有利弊,适合解决不同的工程优化问题。然而,随着新问题的不断产生,它们有可能都不再适用。面对不断产生的复杂问题,群智能算法的主要发展方向可分为两个,其一是设计新的算法去更好地解决问题,其二是对现有群智能技术进行改进以使其在求解问题时性能更佳。

教与学优化算法(teaching-learning-based optimization algorithm, TLBO)^[9]是2011年Rao等基于一个自然班级中老师与学生间的教学行为而提出的一种群智能优化算法。人类是自然界最聪明的生物,因为他们可以通过不断地学习获得丰富的知识和技能,课堂教育是学生获取知识的重要途径。在课堂上,教师通过设计课程、设计教学计划、实施教学活动和反思等一系列教学活动,努力提高班级平均成绩。同时,学生也通过

努力学习和与其他人交流来提高自己的表现,实现整个团队的动态优化。TLBO很好地表征了人类的教学行为,而且具有不需要其他特殊参数、计算量小、一致性高等优势。但是,它依然存在着收敛速度不够快、解的质量不够高、在某些问题上局部最优避免性差等问题。因此,许多学者对其进行了改性研究。例如,为提高TLBO的求解质量以及加快TLBO的收敛速度和运行时间,Li等^[10]于2013年提出了一种改良的教与学优化算法A-TLBO,并利用A-TLBO对最小二乘支持向量机的超参数进行了优化调整及验证;为提升算法性能,于坤杰等^[11]于2014年提出一种基于线性衰减收敛因子和信任权重的改进TLBO;为解决TLBO寻优精度低、稳定性差的问题,闫苗苗等^[12]于2019年设计了一种多班级交互式教学优化算法(multi-classes interaction TLBO, MCITLBO)。然而,这些改性算法的设计要么使得计算成本增高,要么添加了一些需要调整的特殊参数。因此,本研究基于新的优化策略提出一种改进教与学优化算法。

2种改进策略分别是非线性收敛因子调整策略以及标杆管理策略。对于收敛因子,在标准TLBO中,收敛因子是随机取值为1或2,这种策略显然不符合实际,因为,随着教学经验的累积,教师的教学水平也应该得以提升。从算法角度来看,为加快收敛,算法的收敛因子应该随种群所获得的信息量进行动态调整,而不是笼统地设为1或2。此外,标杆管理是一种先进的管理学思想,针对学习阶段随机学习所导致的“无效学习”等问题,在班级中推选精英组,进而建立班级学习标杆,可以更好地帮助班级成员学习到更多“有用”的知识。从算法角度来看,标杆管理策略进一步主导了种群朝着正确的方向进化,从而加快了算法的收敛速度,使得算法有更多的精

力致力于全局最优解的局部开发。为验证改进策略的有效性，基于2种策略的部署组合提出了3种改进教与学优化算法(modified teaching-learning-based optimization algorithm, MTLBO)的变体，并通过11个基准测试函数进行了实验对比。随后，为进一步验证提出算法的性能，使用性能最优的MTLBO与其他著名的优化算法进行了实验验证与分析。

1 教与学优化算法

TLBO的灵感来源主要是课堂上师生之间的教与学行为，其设计思想主要是基于一个典型的优化系统(即教学活动)的构建。在该系统中，TLBO将一个自然班级视为动态演化的种群，老师和学生都可作为种群中的搜索代理，他们的知识水平或考试成绩作为群智能优化算法的适应值。

在TLBO中，以正态分布来表示班级学生的成绩分布情况，如式(1)所示。

$$f(x)=\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{1}$$

式中： σ^2 表示标准差， μ 表示均值。

假设某学校有2位属于不同的2个班级的教师 T_1 和 T_2 ，他们教授的学生水平完全相同。图1描述了 T_1 和 T_2 各自所教授班级学生成绩在经过一段教学活动后的分布情况。在图1中，曲线1、2表示 T_1 、 T_2 授课班级成绩分布， M_1 和 M_2 分别表示该科目的全班平均成绩。可以看出， M_2 明显高于 M_1 ，这说明 T_2 的教学水平更高。

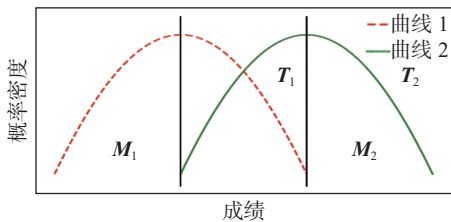


图1 不同教师的教学效果分布图

Fig.1 Distribution of grades obtained by students taught by two different teachers

图2展示了教学活动前、后的班级成绩的变化情况。其中，曲线A和曲线B分别表示在教学活动之前和之后的全班成绩分布， M_A 和 M_B 分别表示教学前后的全班平均成绩， T_A 和 T_B 分别表示教学前后的最佳个体，即班级教师。

假设班级人数为 N ，则其基本种群的数学表示如式(2)所示。

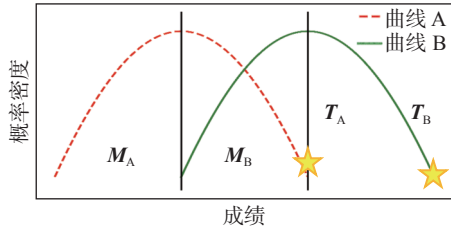


图2 教学活动前后班级学生成绩分布图

Fig.2 Distribution of grades obtained by students taught by a teacher

$$P=\begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ \vdots \\ X_N \end{bmatrix}=\begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,n} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,n} \\ x_{3,1} & x_{3,2} & \cdots & x_{3,n} \\ \vdots & \vdots & & \vdots \\ x_{N,1} & x_{N,2} & \cdots & x_{N,n} \end{bmatrix} \tag{2}$$

式中： $P=[X_1,X_2,X_3,\cdots,X_N]$ 表示班级中的所有学生； $x_{i,j}(i=1,2,3,\cdots,N,j=1,2,3,\cdots,n)$ 表示第 i 个学生在第 j 个科目上所取得的成绩或者所拥有的知识水平。

简单来讲，教与学优化算法可划分为2个阶段：教学阶段与学习阶段。

1.1 教学阶段

在教学阶段，班上最优秀的个体被指定为老师。教师的目标是向所有学生传授知识，并尝试将班级的平均成绩提高到自身水平。假设当前的迭代次数为 k ，并且令 M_k 为所有个体的平均成绩，而 T_k 为迭代次数为 k 时的教师。当前班级平均成绩与教师水平之间的差异

$$D=r_k(T_k-T_F M_k) \tag{3}$$

$$T_F=\begin{cases} 1, & r\leq 0.5 \\ 2, & r> 0.5 \end{cases} \tag{4}$$

式中： T_F 称为教学因子，值为1或2； r_k ， r 表示0~1之间的随机数。

在学习阶段，个体进行位置更新的计算式为

$$X_{new,i}^k=X_i^k+D \tag{5}$$

式中： $X_i^k(i=1,2,\cdots,N)$ ， $X_{new,i}^k$ 分别表示学生位置更新前、后的坐标。

1.2 学习阶段

在学习阶段，学生之间通过两两交流学习来增长知识，交流的方法主要包括小组讨论以及个人经验介绍等。这一阶段具体的算法实施步骤如下：

对个体 X_i^k ，随机挑选另一个体 $X_j^k(i\neq j)$ ，如果 $f(X_i^k)<f(X_j^k)$ ，则 X_i^k 的位置更新公式可表示为式

(6), 否则表示为式 (7)。

$$\mathbf{X}_i^{k+1} = \mathbf{X}_i^{k'} + r_i(\mathbf{X}_i^{k'} - \mathbf{X}_j^{k'}) \quad (6)$$

$$\mathbf{X}_i^{k+1} = \mathbf{X}_i^{k'} + r_i(\mathbf{X}_j^{k'} - \mathbf{X}_i^{k'}) \quad (7)$$

如果计算得到的 $\mathbf{X}_i^{k+1}$ 更优, 则接受位置更新, 否则保持 $\mathbf{X}_i^{k'}$ 不变。

2 改进的教与学优化算法

2.1 非线性收敛因子

在教学阶段, 教师的教学因子 T_F 是一个关键参数, 它可以决定学生个体在多大程度上依据均值 $\mathbf{M}_k$ 进行位置更新。在基础 TLBO 中, T_F 在每一次迭代中等概率地取值为 1 或者 2。然而, 这种随机改变并不能科学地表征群体行为。在设想中, 学生随着学习进程的推进其自主性更强, 随着知识和学习方法的积累其学习能力也会得到相应的强化。在经过大量数值实验之后, 发现 T_F 大于 2 更有利于群体收敛, 算法能够更快地逼近全局最优解。从现实角度来讲, T_F 大于 2 意味着这个班级的教师教学水平和学生学习能力都很强, 这是一个高效的班级, 可以类比学校中的“重点实验班”或“尖子班”。同时, 随着教学活动的不断深化与推进, 教师的教学水平也会相应地得以提升, 而不是一贯地随意分配为 1 或者 2。为更好地表征上述现象, 本文设计了一种随着迭代进行非线性变化的收敛因子, 如式 (8) 所示。

$$T_F = 2 + (k/T)^{1/2} \quad (8)$$

式中: k 表示当前迭代次数; T 表示最大迭代次数。

2.2 标杆管理策略

在传统 TLBO 的学习者阶段, 班级成员通过相互交流和来增加他们的知识。然而, 这种学习方法有一定的缺点。最重要的一点是, 班级成员缺乏优秀个人的指导。这种相互交流和来是杂乱无章的, 没有一定的基准和标准, 因此, 可能会导致很多无用的“学习”。从种群进化的角度来看, 这种无效学习策略的存在会使种群在一定程度上朝着错误的方向进化。

为解决上述问题, 引入了标杆管理的概念。标杆管理又称为基准管理, 是管理学中的一种重要思想, 一般用于企业管理当中。具体地, 标杆管理是指企业将自身情况与行业内龙头企业进行比较, 从而得以借鉴他人先进经验, 改善自身缺

陷^[13-15]。这一理念同样适用于群智能优化算法的种群进化, 对于 TLBO, 场景类似度极高。因此, 将之引入到 TLBO 的学习阶段, 即对学习阶段进行“标杆管理”。

首先, 需要在当前种群中选择合适的标杆个体, 该个体代表着班级中的精英组^[16]所具有的知识, 具体算法设计与建模过程如下: 每次迭代中, 对通过教学阶段的新班级按照个体的成绩(即适应值)进行排序, 选择成绩最好 $\mathbf{X}_s$ 和第二的 $\mathbf{X}_c$ 个体组成精英组。标杆个体由精英组内推选产生, 其为当前班级中成绩最佳的个体。现介绍标杆个体推选的具体实施步骤。

推选精英组, 如式 (9) 所示。对组内个体求均值得到备选标杆 $\mathbf{X}_b$, 在备选标杆和最佳个体之间依适应值最佳原则选出最终标杆。如果 $f(\mathbf{X}_b) < f(\mathbf{X})$, 则选定 $\mathbf{X}_b$ 为最终标杆 $\mathbf{X}_c$; 否则, 选定 $\mathbf{X}_s$ 作为最终标杆 $\mathbf{X}_c$ 。

$$\mathbf{G}_l = [\mathbf{X}_s, \mathbf{X}_c] \quad (9)$$

式中: $\mathbf{G}_l$ 为精英组。

然后, 班级个体依据推选出的标杆进行学习。算法实施中, 其他个体依据标杆进行位置更新的公式如式 (10) 所示。

$$\mathbf{X}_i^{k+1} = \mathbf{X}_i^{k'} + r_i(\mathbf{X}_c - \mathbf{X}_i^{k'}) \quad (10)$$

式中: r_i 表示介于 $[0, 1]$ 之间的随机数, $\mathbf{X}_i^{k'}$ 表示第 i 个体在教学阶段后的位置向量, $\mathbf{X}_i^{k+1}$ 表示第 i 个体在学习阶段经过向标杆学习后的位置向量。

如果计算得到的 $\mathbf{X}_i^{k+1}$ 更优, 则接受位置更新, 否则保持 $\mathbf{X}_i^{k'}$ 不变。

2.3 MTLBO 算法

综合使用改进策略的改进教与学优化算法 (MTLBO) 步骤如下:

- a. 参数初始化。设定种群规模 N , 总迭代次数 T 。
- b. 待解决问题参数空间范围内进行班级种群初始化, 并计算每个个体的适应值。
- c. 令 $t = 1$, 迭代开始。
- d. 教学阶段开始, 计算 $\mathbf{M}_k$, 依据式 (8) 计算 T_F 。
- e. 计算 $\mathbf{D}$, 班级个体依据 $\mathbf{D}$ 更新位置。
- f. 更新班级所有个体适应值, 教学阶段结束。
- g. 学习阶段开始, 推选标杆个体 $\mathbf{X}_c$ 。
- h. 班级个体以 $\mathbf{X}_c$ 为榜样根据式 (10) 更新位置。
- i. 更新班级所有个体适应值, 学习阶段结束。

j. 判断是否满足循环终止条件, 若是, 返回全局最佳解; 否则, 令 $t=t+1$, 返回步骤 d。

3 实验与分析

为验证非线性收敛因子和标杆管理策略的引入对算法性能的影响, 选择 11 个著名的基准测试函数进行 2 组对比实验。在第 1 组对比实验中, 依据对 2 种改进策略的不同部署情况, 构建了 3 种 MTLBO 的变体, 新设计的 MTLBOs 与标准 TLBO 进行了实验对比与分析。在第 2 组对比实验中, 为进一步验证设计策略的有效性, 选择了其他几种应用广泛的群智能优化技术进行对比实验

与分析。所有实验涉及代码均是使用 Python 3.7 编程实现, 且所有实验均在 Window 操作系统下实施。为全面衡量算法性能, 使用了多个维度的基准测试函数进行实验。同时, 为保证测试结果的鲁棒性, 每个测试实验均以 30 次独立运行得到的均值和标准差作为最终的衡量依据。

测试中的 11 个基准测试函数包含 6 个单峰测试函数以及 5 个多峰测试函数, 如表 1 所示。单峰测试函数为 F1~F7, 它们在搜索空间中只有一个严格局部最优值, 因而可以用于测量算法的收敛性; 多峰测试函数为 F8~F12, 它们在搜索空间中往往具有多个局部最优解, 因而可以用于度量算法的局部最优避免能力。

表 1 基准测试函数
Tab.1 Benchmark function

函数	函数表达式	范围	解
F1	$f_1(x) = \sum_{i=1}^n x_i^2$	$[-100,100]$	0
F2	$f_2(x) = \sum_{i=1}^n x_i + \prod_{i=1}^n x_i $	$[-10,10]$	0
F3	$f_3(x) = \sum_{i=1}^n \left(\sum_{j=1}^i x_j \right)^2$	$[-100,100]$	0
F4	$f_4(x) = \max_i \{ x_i , 1 \leq i \leq n \}$	$[-100,100]$	0
F5	$f_5(x) = \sum_{i=1}^{n-1} \left[100(x_i + 1 - x_i^2)^2 + (x_i - 1)^2 \right]$	$[-30,30]$	0
F6	$f_6(x) = \sum_{i=1}^n i x_i^4 + \text{random}[0, 1]$	$[-128,128]$	0
F7	$f_7(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$	$[-5.12,5.12]$	0
F8	$f_8(x) = 20 \exp \left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2} \right) \exp \left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i) \right) + e + 20$	$[-32,32]$	0
F9	$f_9(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos \left(\frac{x_i}{\sqrt{i}} \right) + 1$	$[-600,600]$	0
F10	$f_{10}(x) = \sum_{i=1}^n u(x_i, 10, 100, 4) + \frac{\pi}{n} \left\{ 10 \sin(\pi y_1) + \sum_{i=1}^{n-1} (y_i - 1)^2 \left[1 + 10 \sin^2(\pi y_{i+1}) \right] + (y_n - 1)^2 \right\}$	$[-50,50]$	0
F11	$f_{11}(x) = \sum_{i=1}^n u(x_i, 5, 100, 4) + 0.1 \left\{ \sin^2(3\pi x_1) + \sum_{i=1}^n (x_i - 1)^2 \left[1 + \sin^2(3\pi x_i + 1) \right] + (x_n - 1)^2 [1 + \sin^2(2\pi x_n)] \right\}$	$[-50,50]$	0

3.1 MTLBOs 与标准 TLBO 的比较

对比算法为 TLBO 和 MTLBOs, 这 2 种算法本身不需要其他参数, 但在实验前, 首先要对 2 个群智能优化算法的超级参数进行设定。对于种群尺寸和最大迭代次数, 分别设定为 20, 1000。**表 2** 列出了待比较的 TLBO 和 3 种 MTLBO 的变体 MTLBO1, MTLBO2, 以及 MTLBO3, **表 3** 和**表 4** 分别展示了 4 种对比算法在 11 个 10 维和 30 维基准测试函数上的实验结果, **表 5** 展示了各算法 30 次独立运行中的运行总时间(单位: s)。在**表 3~5** 中, 粗体表示获得的最佳结果。

从**表 3** 和**表 4** 可以看出, 对于函数 F1 和 F3, MTLBO2 和 MTLBO3 在 2 个维度上均获得理论最优解, 同时, 标准差为 0 说明算法在解决问题 F1 和 F3 时的鲁棒性非常高。对于函数 F1 和 F3, MTLBO3 效果最佳, MTLBO2 次之, 标准 TLBO 最差。对于函数 F5, MTLBO1 的结果优于其对比者。对于函数 F6, MTLBO3 和 MTLBO2 的结果排在前两位, 其中, 当维度为 10 时, MTLBO3 略胜一筹; 当维度为 30 时, MTLBO2 的结果要更好。对于函数 F9 和 F11, 所有算法均取得其理论最佳解, 后文将从收敛速度方面进行进一步比较分析。对于函数 F8, MTLBO3 和 MTLBO2 求得更好的数值结果。对于函数 F10, 当维度为 10 时, MTLBO2 结果最优, MTLBO3 次之; 当维度为 30 时, MTLBO3 结果最优, MTLBO2 次之。对于函数 F11, 标准 TLBO 的结果胜过其他对比算法。

从**表 5** 可知, 在不同维度下各算法的计算效果差异明显。当函数维度为 10 时, MTLBO1 在函数 F1, F2, F3, F7 上运算效果更快, MTLBO2 在函数 F4, F6 上计算时间最短, MTLBO3 在函数 F4, F5, F10 和 F11 上优于其他算法, TLBO 则在函数 F8 和 F9 取得最短的运算时间。当函数维度为 30 时, MTLBO1 在函数 F7, F8, F9 上效果最佳, MTLBO3 在函数 F10, F11 上具有最高的计算效果, TLBO 在单峰测试函数 F1—F6 上的运行时间优于其余算法。

根据以上分析, 综合使用前面 2 种改进策略的 MTLBO3 表现最优, 它在大多数问题上获得最佳数值结果。同时, 仅使用标杆管理策略的 MTLBO2 表现颇佳, 仅在少数问题上稍弱于 MTLBO3。综合来看, MTLBO3 和 MTLBO2 明显改进了标准 TLBO 的求解性能。此外, 仅使用非线性收敛因子策略的 MTLBO1 也在大多数问题上

表 2 对比算法及描述

Tab.2 Comparison algorithms and description

算法	算法描述
TLBO	标准教与学优化算法
MTLBO1	使用非线性收敛因子策略改进的教与学优化算法
MTLBO2	使用标杆管理策略改进的教与学优化算法
MTLBO3	使用非线性收敛因子和标杆管理策略改进的教与学优化算法

表 3 TLBO 和 MTLBOs 在 10 维基准函数上的实验结果

Tab.3 Experimental results of TLBO and MTLBOs on 10-dimensional benchmark functions

函数	指标	TLBO	MTLBO1	MTLBO2	MTLBO3
F1	平均值	1.36E-236	4.72E-264	0	0
	标准差	0	0	0	0
F2	平均值	3.00E-117	8.02E-131	3.04E-193	3.31E-299
	标准差	1.12E-116	3.93E-130	0	0
F3	平均值	4.10E-238	1.90E-267	0	0
	标准差	0	0	0	0
F4	平均值	8.72E-112	2.91E-127	5.93E-185	4.35E-296
	标准差	3.06E-111	6.68E-127	0	0
F5	平均值	8.92E+00	8.90E+00	8.98E+00	8.97E+00
	标准差	6.37E-02	4.42E-02	2.09E-02	1.80E-02
F6	平均值	2.36E-04	1.68E-04	8.16E-05	4.72E-05
	标准差	1.38E-04	8.78E-05	5.92E-05	2.83E-05
F7	平均值	0	0	0	0
	标准差	0	0	0	0
F8	平均值	3.23E-15	3.35E-15	-4.44E-16	-4.44E-16
	标准差	6.38E-16	1.28E-15	0	0
F9	平均值	0	0	0	0
	标准差	0	0	0	0
F10	平均值	-2.75E+00	-2.77E+00	-2.11E+00	-2.32E+00
	标准差	2.66E-01	1.52E-01	5.00E-01	3.45E-01
F11	平均值	6.42E-01	6.35E-01	1.13E+00	9.65E-01
	标准差	2.78E-01	2.38E-01	2.19E-01	2.88E-01

胜过标准 TLBO, 尽管从数值上来看提升的幅度不大。从计算效率的角度来看, TLBO 和 3 种 MTLBOs 在不同维度的不同问题上存在一定差异, 当函数维度较低时, MTLBO1 和 MTLBO3 具有更高的运算效率; 当维度较高时, TLBO 在单峰测试函数上的运行时间优于其余算法, MTLBO1 和 MTLBO3 在多峰测试函数上表现颇佳。综上所述, 改进算法在不增加时间成本的前提下改进了 TLBO 的性能, 本文所提出的 2 种策略都是有效的, 尤其是标杆管理策略。

3.2 与其他群智能优化算法的比较

第一组实验主要对 MTLBOs 和 TLBO 进行了实验比较与分析, 基于分析结果, 选择表现最佳的 MTLBO3 进行后续实验。选择一些其他著名的

表 4 TLBO 和 MTLBOs 在 30 维基准函数上的实验结果

Tab.4 Experimental results of TLBO and MTLBOs on 30-dimensional benchmark functions

函数	指标	TLBO	MTLBO1	MTLBO2	MTLBO3
F1	平均值	6.01E-234	2.78E-267	0	0
	标准差	0	0	0	0
F2	平均值	3.24E-119	1.48E-133	5.77E-195	8.52E-297
	标准差	7.18E-119	4.95E-133	0	0
F3	平均值	1.71E-234	6.09E-266	0	0
	标准差	0	0	0	0
F4	平均值	5.88E-114	1.10E-128	2.88E-187	1.18E-296
	标准差	1.44E-113	3.55E-128	0	0
F5	平均值	2.89E+01	2.89E+01	2.90E+01	2.90E+01
	标准差	3.37E-02	3.16E-02	1.46E-02	2.29E-02
F6	平均值	2.05E-04	2.11E-04	5.61E-05	5.63E-05
	标准差	1.20E-04	1.05E-04	4.50E-05	3.56E-05
F7	平均值	0	0	0	0
	标准差	0	0	0	0
F8	平均值	2.87E-15	3.23E-15	-4.44E-16	-4.44E-16
	标准差	1.28E-15	6.38E-16	0	0
F9	平均值	0	0	0	0
	标准差	0	0	0	0
F10	平均值	-2.47E-01	-2.38E-01	1.84E-01	2.68E-03
	标准差	2.07E-01	2.03E-01	2.20E-01	2.56E-01
F11	平均值	3.10E+00	3.24E+00	4.19E+00	3.83E+00
	标准差	5.20E-01	4.69E-01	4.35E-01	3.30E-01

表 5 对比算法运行时间

Tab.5 Comparison of running time for algorithms

函数	维度	TLBO	MTLBO1	MTLBO2	MTLBO3
F1	10	69	52	60	62
	30	89	106	125	148
F2	10	83	70	72	72
	30	157	189	191	189
F3	10	56	49	49	50
	30	129	148	154	150
F4	10	58	56	54	54
	30	123	130	134	134
F5	10	79	71	69	68
	30	251	317	327	321
F6	10	69	62	60	62
	30	138	178	173	166
F7	10	108	105	114	111
	30	284	186	190	191
F8	10	114	119	119	124
	30	299	192	199	199
F9	10	116	122	122	117
	30	320	199	210	207
F10	10	152	153	150	149
	30	481	314	312	310
F11	10	151	150	142	137
	30	441	292	292	288

优化器，如 HS，PSO，MFO 和 GA 实施对比实验。表 6 列出了对比算法和相应的参数设定值。依据表 6 所列，PSO，MFO 和 GA 的种群规模设

为 40，TLBO 和 MTLBOs 的种群规模设为 20，这是因为 TLBO 和 STLBO 在一次迭代中共有 $2N$ (种群规模)次函数评估，而在 PSO，MFO 和 GA 的运行过程中每次迭代仅有 N (种群规模)次函数评估。在这类算法中，算法的函数评估环节是其时间复杂度的主要部分，因为，为保证各对比算法时间复杂度一致，必须要保证各对比算法的函数评估次数一致。另外，除 HS 外其余算法的种群总迭代次数均为 1000。对于 HS，最佳种群规模为 5~7，且在一次迭代中仅有一次函数评估，因此，HS 的种群规模和迭代次数分别设为 40000 和 6。总而言之，为保证无偏性，经过种群规模和总迭代次数的设定，各算法的总函数评估次数均设定为 40000 次，这保证了各对比算法的时间复杂度大致相同。

表 6 对比算法及参数集

Tab.6 Comparison algorithms and parameter sets

算法	种群规模	总迭代次数	其他参数
HS	6	40000	$HMCR=0.9, PAR=0.3, BW=0.1$
PSO	40	1000	$v_{\max} = 20, w_1 = 0.9, w_2 = 0.4, c_1 = 1.2, c_2 = 1.2$
GA	40	1000	$p_c = 0.7, p_m = 0.3$
MFO	40	1000	
TLBO	20	1000	
MTLBO3	20	1000	

如表 6 所示，HS，PSO 以及 GA 这 3 种算法还存在一些特殊参数。对于 HS， $HMCR$ ， PAR 和 BW 分别表示和声记忆库取值概率、音调调整概率和调音带宽，取值分别为 0.9，0.3 和 0.1；对于 PSO， $v_{\max}$ 代表粒子移动的最大速度， w_1 ， w_2 表示初始权重和最终权重， c_1 ， c_2 分别表示认知参数和社会参数，5 种参数的取值依次为 20，0.9，0.4，1.2 和 1.2；对于 GA，选择轮盘赌染色体复制策略、均匀交叉策略以及随机选择变异策略，其中， p_c ， p_m 分别表示染色体交叉概率、变异概率，取值分别为 0.7 和 0.3。以上算法中各个特殊参数的取值均是在原始论文基础上多次实验的经验值。

表 7 列出了对比算法在 40 维测试函数上的实验结果，粗体显示最佳结果。由表 7 可知，MTLBO3 在函数 F1—F4 上远胜过其他对比方法，其次是 TLBO，其中，对于函数 F1 以及 F4，MTLBO3 获

表 7 6 种对比算法在基准函数上的实验结果

Tab.7 Experimental results of six algorithms on the benchmark functions

函数	指标	HS	PSO	GA	MFO	TLBO	MTLBO3
F1	平均值	1.42E+01	1.28E+00	1.13E+01	5.67E+03	6.01E-234	0
	标准差	8.37E+00	3.67E+00	3.88E+00	5.59E+03	0	0
F2	平均值	1.68E-01	7.67E+11	1.32E+00	6.04E+01	3.24E-119	8.52E-297
	标准差	7.81E-02	1.87E+12	1.85E-01	2.49E+01	7.18E-119	0
F3	平均值	5.35E+03	1.85E+04	1.35E+04	3.40E+04	1.71E-234	0
	标准差	1.75E+03	3.00E+04	4.27E+03	1.62E+04	0	0
F4	平均值	8.20E+00	9.34E+00	1.11E+01	2.89E+01	5.88E-114	1.18E-296
	标准差	1.36E+00	1.99E+01	1.96E+00	8.35E+00	1.44E-113	0
F5	平均值	5.06E+02	5.00E+07	5.30E+02	1.33E+07	2.89E+01	2.90E+01
	标准差	4.57E+02	6.17E+07	3.24E+02	2.98E+07	3.37E-02	2.29E-02
F6	平均值	1.19E+03	1.78E+03	6.26E+02	5.73E+08	2.05E-04	5.63E-05
	标准差	9.04E+02	3.76E+03	4.78E+02	8.42E+08	1.20E-04	3.56E-05
F7	平均值	2.02E+00	3.38E+02	5.14E+00	1.75E+02	0	0
	标准差	1.02E+00	1.05E+02	1.34E+00	5.45E+01	0	0
F8	平均值	1.51E+00	1.72E+01	1.80E+00	1.93E+01	2.87E-15	-4.44E-16
	标准差	3.63E-01	3.20E+00	3.07E-01	3.30E+00	1.28E-15	0
F9	平均值	1.12E+00	4.05E-02	1.10E+00	4.52E+01	0	0
	标准差	4.02E-02	4.59E-02	4.38E+02	.50E+01	0	0
F10	平均值	-9.95E-01	4.21E+07	-2.30E-01	8.53E+06	-2.47E-01	2.68E-03
	标准差	2.05E-02	1.30E+08	1.07E-01	4.59E+07	2.07E-01	2.56E-01
F11	平均值	5.39E-01	2.67E+07	4.20E-01	4.10E+07	3.10E+00	3.83E+00
	标准差	2.20E-01	1.44E+08	1.37E-01	1.23E+08	5.20E-01	3.30E-01

得的均值和标准差均为 0，这说明该方法每次实验均可以得到理论最优解，其鲁棒性较强。对于函数 F5 和 F11，TLBO 略胜于 MTLBO3，获得最好的数值结果。对于函数 F6 和 F10，MTLBO3 的数值结果胜过其他对比方法。最后，对于 F7 和 F9，TLBO 和 MTLBO3 均得到理论最优解，优劣无法从数值结果得出，下面将进一步利用收敛曲线进行对比。可以看出，MTLBO3 的数值结果在大多数问题上优于其他对比算法，且在 4 个函数优化问题(F1, F3, F7 以及 F9)上 1000 次迭代内均收敛到理论最佳解。此外，MTLBO3 在大多数问题上具有最小的方差，这说明算法性能稳定，鲁棒性较好。

HS, PSO, GA, MFO, TLBO 以及 MTLBO3 这 6 种算法在 11 个测试函数上 30 次独立运行的总时间列于表 8。由表 8 可知，TLBO 和 MTLBO 的运行时间明显优于其他算法。具体地，TLBO 在函数 F1, F2, F3, F6 以及 F8 上取得最短的运行时间，而 MTLBO3 则在函数 F4, F5, F7, F9, F10 以及 F11 具有最高的计算效率。总体上讲，MTLBO3

和 TLBO 计算效率差异不大，MTLBO3 略胜一筹，同时前者在计算精度上优于后者。

为进一步对比算法性能，画出了 6 种对比算法在部分基准测试函数上的收敛曲线，如图 3 所

表 8 6 种对比算法在 11 个测试函数上的运行时间

Tab.8 Running time of 6 comparison algorithms on 11 test functions

函数	HS	PSO	GA	MFO	TLBO	MTLBO3
F1	819	596	693	677	179	186
F2	848	634	736	707	203	211
F3	831	608	712	686	181	190
F4	770	591	650	652	148	110
F5	736	745	726	736	177	175
F6	601	482	541	498	87	93
F7	757	545	670	651	195	82
F8	779	618	687	694	204	231
F9	736	645	672	678	214	207
F10	753	789	824	858	342	209
F11	640	782	726	768	332	222

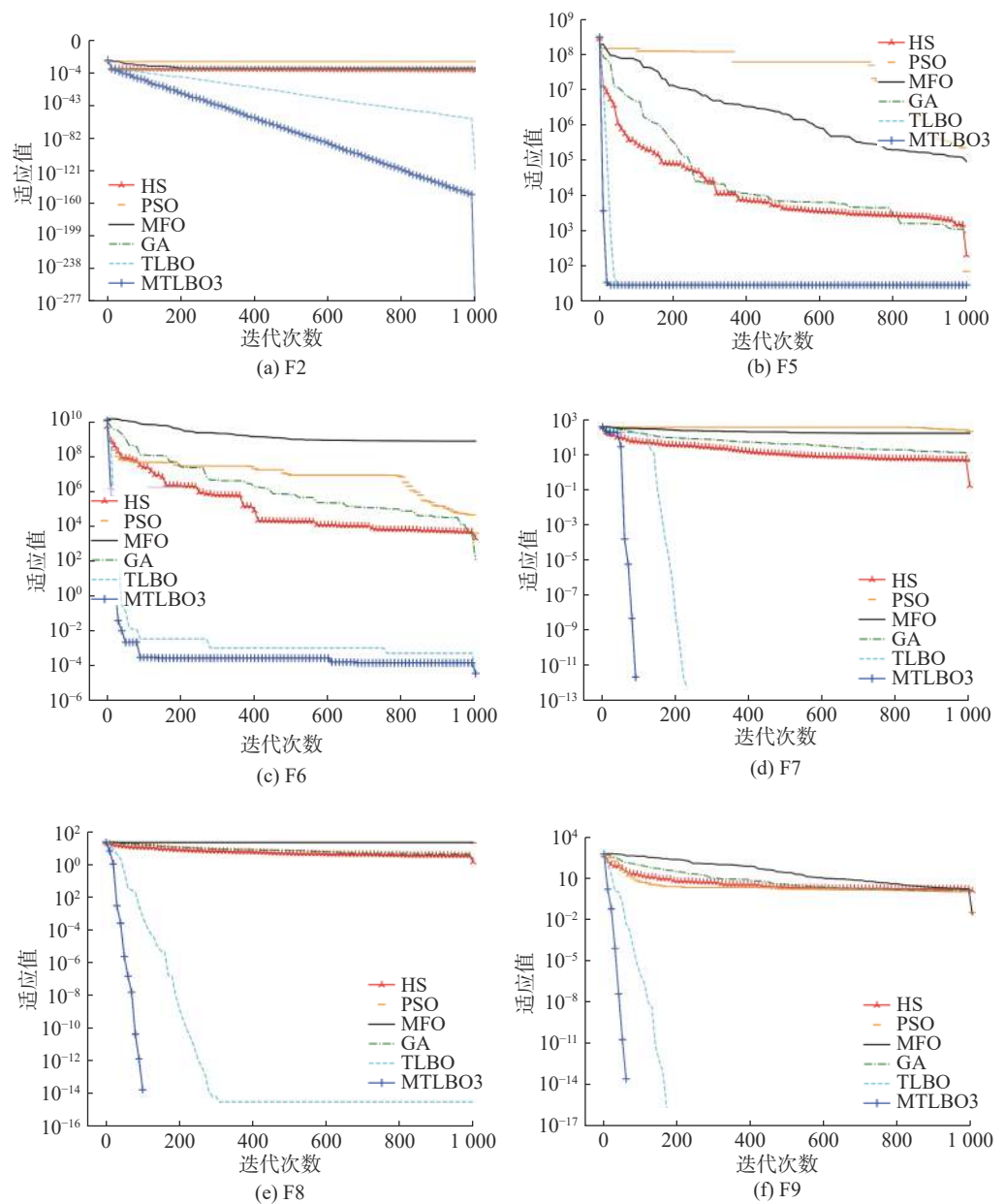


图3 6种比较算法在部分测试函数上的收敛曲线

Fig.3 Convergence curves of 6 comparison algorithms on some test functions

示。可以看出，MTLBO3 在所有问题上均具有最高的收敛速度。综合表 7 中的数值结果，尽管在函数 F5 上 TLBO 数值结果更优，但是，MTLBO3 的收敛速度更快。此外，在函数 F7 和 F9 上，MTLBO3 的收敛速度都快于 TLBO。综上所述，MTLBO 明显改进了标准 TLBO 的搜寻性能。

3.3 约束优化问题

由于函数 F1—F11 均为无约束优化问题，为进一步验证本文所设计策略的有效性，使用一个约束优化问题——拉伸/压缩弹簧设计优化问题进行实验。该优化问题的目的是弹簧质量的最小

值，其中，包含线径 d ，平均线圈直径 D 和有效线圈数 N 这 3 个设计变量，以及与剪应力、弹簧颤动频率和最小挠度相关 4 个约束条件。拉伸/压缩弹簧设计优化问题的物理模型如图 4 所示，数学模型如下：



图4 拉伸/压缩弹簧设计优化问题

Fig. 4 Tension/compression spring design problem

设计变量

$$x=[x_1,x_2,x_3]=[d,D',N']$$

优化函数

$$f(x)=(x_3+2)x_2x_1^2$$

约束条件

$$g_1(x)=1-\frac{x_2^3x_3}{71\,785x_1^4}\leqslant 0$$

$$g_2(x)=\frac{4x_2^2-x_1x_2}{12\,566(x_2x_1^3-x_1^4)}+\frac{1}{5\,108x_1^2}-1\leqslant 0$$

$$g_3(x)=1-\frac{140.45x_1}{x_2^2x_3}\leqslant 0$$

$$g_4(x)=\frac{x_1+x_2}{1.5}-1\leqslant 0$$

设计变量范围

$$0.05\leqslant x_1\leqslant 2.00,\,0.25\leqslant x_2\leqslant 1.30,\,2.00\leqslant x_3\leqslant 15.0$$

所有算法在 30 次独立运行中所得到的最佳实验结果如表 9 所示。其中, 算法 HS, PSO, GA, MFO 以及 MHHO 的实验结果直接来源于文献 [17]。由表 9 可知, 基于非线性收敛因子策略的 MTLBO1 的结果优于其他对比算法, MTLBO2 和 MTLBO3 的实验结果却要差于原始 TLBO。因此, 在使用教与学优化算法解决拉伸/压缩弹簧设计优化问题时, 非线性收敛因子策略依然是一种有效的策略, 而标杆管理策略不再适用。综上可知, 在使用教与学优化算法求解无约束优化问题时, 设计的 2 种策略均是有效的, 特别是标杆管理策略效果显著, 而在求解无约束优化问题时, 仅有非线性收敛因子策略有效。

表 9 拉伸/压缩弹簧问题结果对比

Tab.9 Comparison of results for tension/compression spring problem

算法	<i>d</i> /cm	<i>D</i> '/cm	<i>N</i> '	最优质量/g
HS	0.051 154 00	0.349 871 000	12.076 432 00	0.012 670 60
PSO	0.051 728 00	0.357 644 000	11.244 543 00	0.012 674 70
GA	0.051 480 00	0.351 661 000	11.632 201 00	0.012 704 80
MFO	0.051 994 46	0.364 109 320	10.868 421 86	0.012 666 90
MHHO	0.051 654 87	0.355 881 658	11.338 833 00	0.012 666 19
TLBO	0.051 723 648	0.357 550 372	11.240 318	0.012 665 255
MTLBO1	0.051 688 903	0.356 713 931	11.289 189 06	0.012 665 233
MTLBO2	0.054 686 665	0.433 032 145	7.915 232 505	0.012 840 618
MTLBO3	0.057 319 934	0.507 153 848	5.948 835 74	0.013 245 081

4 结束语

针对标准 TLBO 寻优精度不高、局部最优避免性差以及收敛速度较慢的问题, 本文在 TLBO 的基础上引入 2 种改进策略: 非线性收敛因子调整策略以及标杆管理策略。随后, 基于 2 种策略的数学组合设计了 3 种 MTLBO 的变体, 为验证改进算法的性能, 通过一组函数优化问题与标准 TLBO 进行了实验对比与分析。结果显示, 3 种

MTLBO 都胜过标准 TLBO, 其中, 引入 2 种改进策略的 MTLBO3 表现最优, 其很好地平衡了算法的全局搜索性和局部开发性, 数值结果远胜于标准 TLBO。为增强说服力, 使用著名的群智能优化算法与 MTLBO3 进行了实验对比与分析。数值结果、计算效率和收敛曲线表明, MTLBO3 在寻优精度、局部最优避免性以及收敛速度等性能上优于其他对比算法。最后, 基于工程设计约束优化问题进一步验证 2 种策略的有效性。结果表明,

在无约束优化问题中,非线性收敛因子调整策略依然能改进 TLBO 的性能,而标杆管理策略可能不再适用。

参考文献:

- [1] MÜHLENBEIN H, SCHLIERKAMP-VOOSEN D. Predictive models for the breeder genetic algorithm I. continuous parameter optimization[J]. *Evolutionary Computation*, 1993, 1(1): 25–49.
- [2] EBERHART R C, SHI Y H. Guest editorial special issue on particle swarm optimization[J]. *IEEE Transactions on Evolutionary Computation*, 2004, 8(3): 201–203.
- [3] LEE K S, GEEM Z W. A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice[J]. *Computer Methods in Applied Mechanics and Engineering*, 2005, 194(36/38): 3902–3933.
- [4] KARABOGA D, BASTURK B. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm[J]. *Journal of Global Optimization*, 2007, 39(3): 459–471.
- [5] RASHEDI E, NEZAMABADI-POUR H, SARYAZDI S. GSA: a gravitational search algorithm[J]. *Information Sciences*, 2009, 179(13): 2232–2248.
- [6] MIRJALILI S. Moth-flame optimization algorithm: a novel nature-inspired heuristic paradigm[J]. *Knowledge-Based Systems*, 2015, 89: 228–249.
- [7] HEIDARI A A, MIRJALILI S, FARIS H, et al. Harris hawks optimization: algorithm and applications[J]. *Future Generation Computer Systems*, 2019, 97: 849–872.
- [8] MIRJALILI S, MIRJALILI S M, LEWIS A. Grey wolf optimizer[J]. *Advances in Engineering Software*, 2014, 69: 46–61.
- [9] 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.
- [10] LI G Q, NIU P F, ZHANG W P, et al. Model NO_x emissions by least squares support vector machine with tuning based on ameliorated teaching-learning-based optimization[J]. *Chemometrics and Intelligent Laboratory Systems*, 2013, 126: 11–20.
- [11] 于坤杰, 王昕, 王振雷. 改进的教学优化算法及其应用[J]. *化工进展*, 2014, 33(4): 850–854.
- [12] 闫苗苗, 刘三阳. 多班级交互式教学优化算法[J]. *计算机工程与应用*, 2019, 55(9): 10–16, 64.
- [13] 孔杰, 程寨华. 标杆管理理论述评[J]. *东北财经大学学报*, 2004(2): 3–7.
- [14] 谢安世, 于永达, 黄思明. 一种基于标杆管理的优化算法[J]. *软件学报*, 2014, 25(5): 953–969.
- [15] 王楚君, 许治, 陈丽玉. 基于标杆管理的中国研究型大学科技成果转化效率评价——网络排序方法的运用[J]. *科研管理*, 2020, 41(3): 183–193.
- [16] 于坤杰, 王昕, 王振雷. 基于反馈的精英教学优化算法[J]. *自动化学报*, 2014, 40(9): 1976–1983.
- [17] ZHANG Y, ZHOU X Z, SHIH P C. Modified Harris hawks optimization algorithm for global optimization problems[J]. *Arabian Journal for Science and Engineering*, 2020, 45(12): 10949–10974.

(编辑: 石 瑛)