

摘 要

随着社会经济的发展,人们所面临的优化问题越来越复杂,智能优化算法的性能也面临了巨大挑战。虽然,很多不同类型的改进算法相继被提出,但其性能还存在一定不足,例如:收敛速度较慢、寻优精度较差等。导致这种不足的主要原因在于,很多相关工作是基于算法的历史经验来进行算法设计,而忽略了问题特征对算法性能的影响。为此,本文从问题特征的角度来研究智能优化算法的设计,通过结合两种离线的地形分析技术:信息地形和信息熵,提出了一种在线的混合地形分析技术,能够在算法的执行过程中实时分析问题的地形特征来调整算法的搜索行为,有力弥补了算法设计和问题特征之间的鸿沟。本文主要工作和贡献如下:

(1) 提出的在线混合地形分析技术比现有相关方法更加高效,且时间复杂度更低。为验证这一点,我们与现有的两个相关工作进行了对比,分别是:基于在线离散度量技术的人工蜂群算法、以及基于适应度-距离相关性分析技术的差分进化算法,设计了基于在线混合地形分析技术的对比算法。在 CEC 2013 和 CEC 2014 两套测试函数集上进行实验,实验结果表明对比算法比原算法在结果精度和运行时间上均有更好表现。

(2) 提出的在线混合地形分析技术可用于改善多策略机制,比基于历史经验的多策略机制更加高效。为验证这一点,我们以一项代表性的相关工作为例:多策略人工蜂群算法 (MEABC),设计了相应的对比算法:基于在线混合地形分析技术的多策略人工蜂群算法,从问题特征的角度来修改 MEABC 的多策略机制。在 CEC 2013 和 CEC 2014 两套测试函数集上进行实验,实验结果表明对比算法在结果精度上要优于原算法。

(3) 提出的在线混合地形分析技术可用于改善算法集成机制,比基于历史经验的算法集成机制更加高效。为验证这一点,我们以一项代表性的相关工作为例:集成式差分进化算法 (EDEV),设计了相应的对比算法:基于在线混合地形分析技术的集成式差分进化算法,从问题特征的角度来修改 EDEV 的算法集成机制。在 CEC 2005 和 CEC 2017 两套测试函数集上进行实验,实验结果表明对比算法同样在结果精度上要优于原算法。

关键词: 人工蜂群算法; 差分进化算法; 集成; 在线混合地形分析技术; 适应度地形

Abstract

With the development of social economy, people are facing more and more complex optimization problems, and the performance of intelligent optimization algorithms is facing great challenges. Although, many different types of improved algorithms have been proposed one after another, their performance still has certain shortcomings, such as slower convergence speed and poorer optimization-seeking accuracy. The main reason for such shortcomings is that many related works are based on the historical experience of algorithms for algorithm design, while ignoring the impact of problem fitness landscape features on algorithm performance. To this end, this paper studies the design of intelligent optimization algorithms from the perspective of problem fitness landscape features, and proposes an online hybrid landscape analysis technology by combining two offline fitness landscape analysis techniques: information landscape and information entropy, which can analyze the problem fitness landscape features to adjust the search behavior of the algorithm in real time during the execution of the algorithm, and strongly bridges the gap between algorithm design and problem fitness landscape features. The main work and contributions of this paper are as follows:

(1) The proposed online hybrid landscape analysis technology is more efficient and has lower time complexity compared to existing related methods. To verify this, we compared it with two existing related works: the artificial bee colony algorithm based on online dispersion Metric technology, and the differential evolution algorithm based on fitness-distance correlation analysis technology, and designed a comparative algorithm based on online hybrid landscape analysis technology. Experiments were conducted on two sets of test functions, CEC 2013 and CEC 2014, and the results showed that the comparative algorithm had better solving performance and less running time than the original algorithm.

(2) The proposed online hybrid landscape analysis technology can be used to improve multi-strategy mechanisms and is more efficient than multi-strategy mechanisms based on historical empirical information. To verify this, we took a representative work, the Multi-strategy ensemble artificial bee colony algorithm (MEABC), and designed a corresponding comparative algorithm: the multi-strategy artificial bee colony algorithm based on online hybrid landscape analysis technology,

which modifies the multi-strategy mechanism of MEABC from the perspective of problem features. Experiments were conducted on two sets of test functions, CEC 2013 and CEC 2014, and the results showed that the comparative algorithm outperformed the original algorithm in terms of result accuracy.

(3) The proposed online hybrid landscape analysis technology can be used to improve algorithm integration mechanisms and is more efficient than integration mechanisms based on historical empirical information. To verify this, we took a representative work, the ensemble of differential evolution variants (EDEV), and designed a corresponding comparative algorithm: the evolutionary differential evolution algorithm based on online hybrid landscape analysis technology, which modifies the algorithm integration mechanism of EDEV from the perspective of problem features. Experiments were conducted on two sets of test functions, CEC 2005 and CEC 2017, and the results showed that the comparative algorithm also outperformed the original algorithm in terms of result accuracy.

Key words: Artificial bee colony algorithm; Differential evolution algorithm; Ensemble; Online hybrid landscape analysis technology; Fitness landscape

目 录

摘 要	I
Abstract.....	II
目 录	IV
第一章 绪论	1
1.1 研究背景.....	1
1.2 智能优化算法.....	2
1.2.1 基本人工蜂群算法.....	2
1.2.2 基本差分进化算法.....	5
1.3 适应度地形分析技术.....	7
1.4 本文主要工作及组织结构.....	8
第二章 在线混合地形分析技术	10
2.1 代表性的适应度地形分析技术.....	10
2.1.1 适应度-距离相关性	10
2.1.2 离散度量.....	10
2.1.3 信息地形.....	11
2.1.4 信息熵.....	11
2.2 本文提出的在线混合地形分析技术.....	12
2.3 本章小结.....	14
第三章 基于在线混合地形分析技术的人工蜂群算法	15
3.1 引言.....	15
3.2 FLABC 算法.....	15
3.3 FLABC-LM 算法	17
3.4 实验验证.....	17
3.4.1 实验设置.....	18
3.4.2 参数敏感性分析.....	20
3.4.3 结果精度对比.....	21
3.4.4 策略有效性验证.....	23
3.4.5 运行时间对比.....	25
3.5 本章小结.....	26
第四章 基于在线混合地形分析技术的差分进化算法	27

4.1 引言.....	27
4.2 DEET 算法.....	27
4.3 DEET-LM 算法.....	29
4.4 实验验证.....	30
4.4.1 实验设置.....	30
4.4.2 参数敏感性分析.....	30
4.4.3 结果精度对比.....	32
4.4.4 策略有效性验证.....	34
4.4.5 运行时间对比.....	35
4.5 本章小结.....	36
第五章 基于在线混合地形分析技术的多策略人工蜂群 算法.....	37
5.1 引言.....	37
5.2 MEABC 算法.....	37
5.3 MEABC-LM 算法.....	38
5.4 实验验证.....	40
5.4.1 实验设置.....	40
5.4.2 参数敏感性分析.....	40
5.4.3 结果精度对比.....	41
5.4.4 策略有效性验证.....	43
5.4.5 运行时间对比.....	45
5.5 本章小结.....	45
第六章 基于在线混合地形分析技术的集成式差分进化 算法.....	46
6.1 引言.....	46
6.2 EDEV 算法.....	46
6.3 EDEV-LM 算法.....	47
6.4 实验验证.....	48
6.4.1 实验设置.....	48
6.4.2 参数敏感性分析.....	48
6.4.3 结果精度对比.....	49
6.4.4 与三种改进的算法对比.....	51
6.4.5 与其他智能优化算法对比.....	53
6.5 本章小结.....	56
第七章 总结与展望.....	57
7.1 论文总结.....	57

7.2 工作展望.....	58
参考文献	59
致 谢	64
在期间公开发表论文（著）及科研情况	65

第一章 绪论

1.1 研究背景

工程领域存在许多实际的优化问题，如工程设计^[1,2]、资源分配^[3,4]和生产调度^[5,6]。工程设计是通过最小化成本或提高效率来优化，这类问题通常包括设计参数优化和性能优化等方面；资源分配优化是在资源受限的情况下，使得生产或项目可以最大化地使用可用的资源，这类问题通常涉及制定计划以最大化利用资源；生产调度优化是通过管理和调度生产活动，以最大限度地提高生产效率、降低成本和提高质量的过程。这类问题通常涉及制定生产计划。随着社会的发展，人们对复杂优化问题的解决需求越来越高，传统的数学优化算法需要基于数学模型或问题的特定性质来解决问题，但这种方法并不总是可行或高效。因此，需要寻找更加高效、精确、鲁棒性更强的算法。相对于传统的数学优化算法，智能优化算法在某些方面具有更加优越的性能，能够解决非常复杂的非线性、非凸、多峰和离散的优化问题。智能优化算法是指一类基于启发式搜索策略的优化算法，包含了粒子群算法^[7]（简称PSO）、遗传算法^[8]（简称GA）、差分进化算法^[9]（简称DE）、人工蜂群算法^[10]（简称ABC）等，智能优化算法被用于许多领域^[11-13]。智能优化算法作为一种有效解决优化问题的方法，具有广泛的应用前景和研究价值。

大多数研究工作都是从算法本身出发进行设计和改进，较少考虑问题的适应度地形结构特征。在工程实践中发现，现有的智能优化算法在求解复杂问题的有效性越来越低。其中的客观原因是新的优化需求不断地提出，优化的问题越来越复杂，难以求解。现有的智能优化算法难以满足实际应用中的需求，这里面存在着两个主观因素。其一是从算法自身的角度设计，无法适应新的优化问题，算法性能不够高效；其二，算法使用了固定解生成策略及参数值，对不同的问题的处理缺乏灵活性，没有考虑问题的适应度地形结构特征信息。这使得智能优化算法在寻求新解时，会呈现收敛速度太慢，求解性能不够高效。因此本文将从问题的适应度地形结构特征角度考虑，对智能优化算法进行改进。

为了更好的理解问题的特性，研究学者提出大量的适应度地形分析技术^[14-20]，用来刻画问题的适应度地形结构，达到量化问题复杂程度的目的。现有的适应度地形分析技术，需要大规模地采样，导致其只能采用离线方式计算，很难应用到算法的设计中。因此将适应度地形分析技术以在线方式应用到算法设计，显得尤为重要。其中 Zhou^[21]首次采用离散度量应用到人工蜂群算法，Li^[22]采用适

适应度-距离相关性分析种群进化的状态，以上两种适应度地形分析技术，从问题的适应度地形结构特征角度考虑，能使算法的性能更加高效。

大多数现有的算法从算法本身设计，求解性能不够高效，但是也有少量的算法从问题的适应度地形结构特征角度设计。即使从问题的适应度地形结构特征角度考虑，例如上述提到两个适应度地形分析技术应用到算法的设计，其存在一些不足之处，首先适应度地形分析技术刻画问题的特征可能不太准确，影响算法的求解性能；其次适应度地形分析技术计算十分耗时，需要消耗大量的计算资源。为了解决上述的问题，本文提出一种在线混合地形分析技术，通过分析问题的结构特征，以此指导算法的设计。

1.2 智能优化算法

本文主要选择两种代表性的智能优化算法开展了具体的研究工作，分别为人工蜂群算法和差分进化算法，以下对这两种算法进行简要描述算法流程，以及介绍相关工作。

1.2.1 基本人工蜂群算法

简单概述为，ABC 将问题的求解过程模拟成蜜蜂采蜜行为，蜂群根据其职责划分为三类分别为雇佣蜂、观察蜂和侦察蜂。每种类型的蜜蜂有明确的任务，首先雇佣蜂在整个搜索空间中探索潜在区域，在已经发现的食物源中选择质量更好的食物源，并且以一定几率以舞蹈的形式，把食物源的位置和信息传递给在蜂房等待中的观察蜂；其次观察蜂有偏好的选择蜜量较多的食物源，之后在该优质的食物源附近继续寻找新的食物源，蜜量愈大，则表明该食物源附近发现更好的食物源可能性愈大，观察蜂会向更优质的食物源靠拢。当某个食物源的质量得不到改善的情况下，在此处的观察蜂就会成为侦察蜂，然后抛弃该食物源并在搜索空间中随机选择位置搜索新的食物源。有关如何决定是否舍弃某个食物源，ABC 设置了一个阈值 *limit*。以下详细介绍 ABC 算法框架进行详细描述：

● 初始化阶段

ABC 算法首先用公式(1-1)初始化种群 *POP*，包含 *SN* 个体（食物源的信息）。

$$x_{i,j} = LOW_j + rand_{i,j} \cdot (UP_j - LOW_j) \quad (1-1)$$

其中， LOW_j 和 UP_j 分别表示优化问题搜索空间第 $j(j \in \{1,2, \dots, d\})$ 维的下界和上界； d 表示优化问题的维度； $rand_{i,j}$ 表示第 i 个体 j 维的一个随机数,取值范围为 $[0,1]$ ； x_i 表示第 i 个体，之后 ABC 将会执行一个循环，即雇佣蜂、观察蜂和侦察蜂三个阶段，按先后顺序重复执行，直至满足终止条件。

● 雇佣蜂阶段

当种群进行迭代时，算法根据解搜索方程搜索新的食物源，种群中每个个体利用公式(1-2)生成候选个体。

$$v_{i,jd} = x_{i,jd} + \phi_{i,jd} \cdot (x_{i,jd} - x_{r,jd}) \quad (1-2)$$

其中， jd 表示随机抽取的维度； $x_{r,jd}$ 是随机抽取的个体，不同于 x_i 且满足 $x_{r,jd} \neq x_{i,jd}$ ； v_i 则是 x_i 生成的子代个体，即候选个体； $\phi_{i,jd}$ 是 $[-1,1]$ 是服从正态分布的随机数。为了方便构建食物选择概率模型，需由公式(1-3)计算出个体的适应度值，适应度越大则表明食物源质量越高。

$$fit(x_i) = \begin{cases} \frac{1}{1+func(x_i)} & \text{if } obj(X_i) \geq 0 \\ 1 + abs(func(x_i)) & \text{otherwise} \end{cases} \quad (1-3)$$

其中， $func(x_i)$ 表示 x_i 的目标函数值； abs 是绝对值函数； $fit(x_i)$ 代表了 x_i 的适应度值；针对最小化优化问题，目标函数值愈小，它的适应度值愈大，代表解的品质愈好。按照贪婪选择方式，个体 x_i 和 v_i 的适应度值较高，将被选择进入下一代，具体公式(1-4)如下。

$$x'_i = \begin{cases} x_i & \text{if } fit(x_i) \geq fit(v_i) \\ v_i & \text{otherwise} \end{cases} \quad (1-4)$$

其中， x'_i 表示能够进入下一代的个体。之后进入观察蜂阶段。

● 观察蜂阶段

ABC 首先用公式(1-5)计算每一个体被选择进入观察蜂阶段的概率。

$$p_i = \frac{fit(x_i)}{\sum_{n=1}^{SN} fit(x_n)} \quad (1-5)$$

其中， p_i 代表个体 x_i 的概率。可见，个体的适应度值愈大，被选择的概率愈高。因此，较优质的食物源可能进行多次重复搜索，而较好个体的信息指导算法的搜索，确保了收敛性。

计算全部个体的概率后，通过公式(1-2)生成 SN 候选个体，并通过公式(1-4)进行贪婪选择，确定进入下一代的个体。

● 侦察蜂阶段

最后，为每个个体设定一个计数器 $trial_i$ ，其记录了个体 x_i 不断地产生候选个体 v_i 的适应度值比 x_i 低的次数。当 $trial_i$ 超过阈值 $limit$ 时，此时个体将会被抛弃，然后利用公式(1-1)重新产生新的个体。

为了更好的表述基本 ABC 算法的执行流程，图 1-1 展示了基本 ABC 的流程图。其中， FES 表示已使用的函数评估次数； $MaxFES$ 则表示算法所允许的最大评估次数，即终止条件。

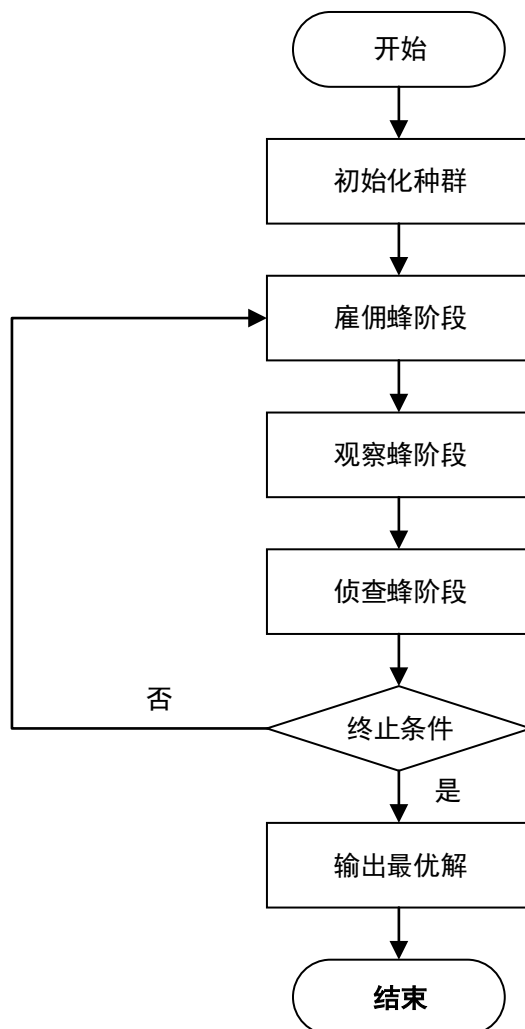


图 1-1 基本 ABC 流程图

解搜索方程对于 ABC 算法的性能有着至关重要的影响，固定的解搜索方程并不一定适用于各种优化问题。基本 ABC 的解搜索方程随机性强，具有较好的勘探能力，在一些较复杂问题能维持多样性，不容易陷入局部最优，但即便勘探到有希望的区域，因其开采能力弱，算法的收敛性较差。Zhu^[23]提出了一种全局最优个体引导的 GABC，利用当前中最优个体和随机个体之间的信息来指导算法搜索，使勘探与开采能力达到平衡。Gao^[24]提出 MABC，以种群的最优个体作为基向量，指导算法的搜索，利用最优个体的信息，有利于增强算法开采能力。Wang^[25]提出 MEABC，集成三种解搜索方程，基于历史成功更新经验信息，为解搜索方程分配计算资源。Kiran^[26]提出 ABCVSS 对每一种解搜索方程都设置计数器，用于统计每一种解搜索方程的成功更新频率，成功更新频率越高的解搜索方程，被选择的概率则越大。Xiang^[27]提出基于引力模型的人工蜂群算法 ABCG，在观察蜂阶段构建了策略池，集成了随机引导搜索、反向学习以及初始化三种解搜索方程。Zhou^[28]提出 MABC-FG，为了达到平衡勘探和开采目的，将个体适应

度排序，然后将种群分为三个组，根据每一组的特征选择对应的解搜索方程。Cui^[29]提出 MPGABC，以兼顾勘探与开采，根据 GABC 的解搜索方程，设计成两种新的解搜索方程，然后通过参数 P 来控制自适应选择解搜索方程。

1.2.2 基本差分进化算法

DE 是较为流行的一种算法，与其他智能优化算法的结构类似，包含种群初始化、变异操作、交叉操作、以及选择操作。以下详细介绍 DE 算法的框架：

- 种群初始化

假设种群大小为 NP ，种群个体的下标取值为 $\{1, 2, \dots, NP\}$ 。所有种群个体根据公式(1-1)在搜索空间中生成。

- 变异

算法的变异主要是通过个体生成一个对应的变异个体实现的。经典的 DE 变异算子有很多种类型^[30]，下文给出常见的五种 DE 变异算子：

(1) DE/best/1:

$$V_i = X_{best} + F \cdot (X_{r1} - X_{r2}) \quad (1-6)$$

(2) DE/rand/1:

$$V_i = X_{r1} + F \cdot (X_{r2} - X_{r3}) \quad (1-7)$$

(3) DE/rand/2:

$$V_i = X_{r1} + F \cdot (X_{r2} - X_{r3}) + F \cdot (X_{r4} - X_{r5}) \quad (1-8)$$

(4) DE/current-to-best/1:

$$V_i = X_i + F \cdot (X_{best} - X_i) + F \cdot (X_{r1} - X_{r2}) \quad (1-9)$$

(5) DE/best/2:

$$V_i = X_{best} + F \cdot (X_{r1} - X_{r2}) + F \cdot (X_{r3} - X_{r4}) \quad (1-10)$$

式中： X_{r1} 、 X_{r2} 、 X_{r3} 、 X_{r4} 、 X_{r5} 是从种群中随机选择互不相同的个体，并且与目标个体 X_i 不相同， X_{best} 表示当前种群中适应度最好的个体。参数 F 是缩放系数，用于控制差分向量的步长，其值常取自于区间 $[0,1]$ ，较大的 F 值可增强算法的勘探能力，而较小的 F 值有助于增强开采能力。

- 交叉

变异操作后，个体 X_i 与变异个体 V_i 通过重组得到试验个体 U_i ，贪婪选择操作如下：

$$U_{i,j} = \begin{cases} V_{i,j} & \text{if } rand \leq CR \text{ or } j = j_{rand} \\ X_{i,j} & \text{otherwise} \end{cases} \quad (1-11)$$

其中， j_{rand} 是一个从 $\{1, 2, \dots, D\}$ 中等概率随机选择的下标， D 为问题的维数， j_{rand} 的存在保证试验个体 U_i 中至少有一个分量是从 V_i 上继承。 CR 为交叉概率，

用于确定试验向量继承父代个体中分量的概率。由于上述交叉由两个向量在相同位上进行分量选择操作，称为二项式交叉算子，二项式交叉算子是使用最多的交叉算子。

● 选择

选择个体 X_i 与其对应的试验个体 U_i 中更优秀的个体作为子代个体。本文解决的优化问题为最小化问题，因此算法的目标为求解优化问题的最小值，具体选择操作如下：

$$X_i = \begin{cases} U_i & \text{if } f(U_i) < f(X_i) \\ X_i & \text{otherwise} \end{cases} \quad (1-12)$$

其中， $f(X_i)$ 是解 X_i 的适应度值评估函数。若根据公式(1-12)， U_i 被选入子代种群，则 X_i 进行更新。

种群初始化以后，执行变异操作、交叉操作和选择操作，并进入循环迭代，直到算法满足终止条件，最终输出最优解。

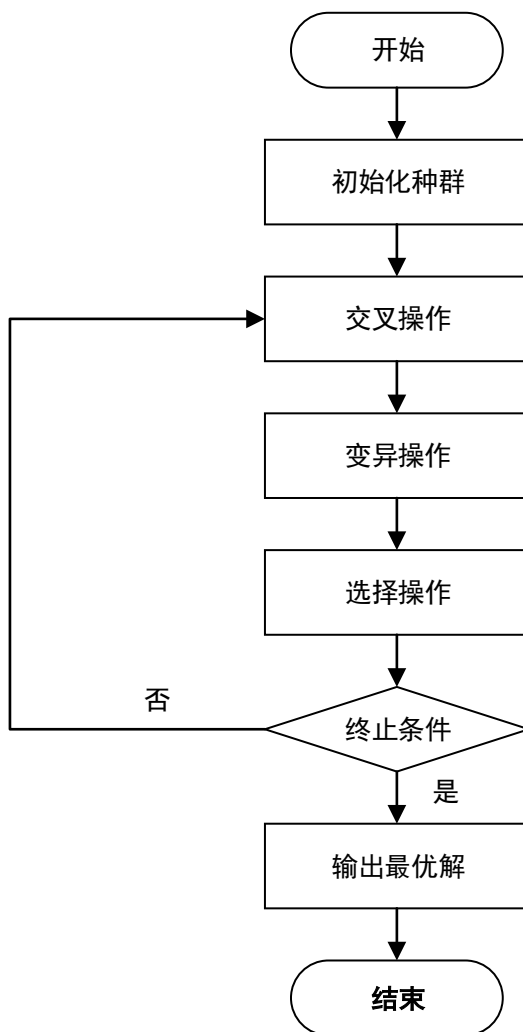


图 1-2 基本 DE 流程图

DE 算法通常使用了固定解生成策略及参数值, 对不同的问题的处理缺乏灵活性, 因此有部分研究学者设计多策略机制。Qin^[31]提出一种自适应差分进化算法, 其中试验向量的生成策略以及参数 F 和 CR , 通过历史经验信息自适应学习。Wang^[32]提出一种 CoDE, 算法融合了 3 种具有不同特点的变异策略和对应的 3 个不同参数组合, 随机结合方式, 用来产生 3 个试验个体, 以贪婪选择方式, 再从其中挑选出最好的个体作为子代个体。Mallipeddi^[33]提出变异策略和参数集成的 DE 变体 EPSDE, 记录变异策略成功更新次数, 并在连续几代中更高概率被选择。Zhang^[34]提出一种外部存档的差分进化算法 JADE, 设计了新的外部存档变异策略和自适应更新控制参数。Wu^[35]提出 MPEDE 设计子种群框架自适应选择变异策略, 由“DE/current-to-pbest/1”和“DE/current-to-rand/1”和“DE/rand/1”构建策略池。设置三个大小相同的较小指标子种群和一个更大的奖励子种群。间隔一定代数, 将根据适应度提高与消耗函数评价次数的比值确定当前性能最好的变异策略。然后奖励子种群动态分配最佳性能的变异策略。Wu^[36]的另一项工作 EDEV 集成三个算法 CoDE、JADE、EPSDE, 将根据适应度提高与消耗函数评价次数的比值确定当前性能最好的算法。然后奖励子种群动态分配最佳性能的算法。Zhou^[37]提出一种多策略算法 MSDE-NS, 与邻域搜索相结合, 依据个体适应度值进行排序, 将种群划分为三个子种群, 各子种群采取不同变异策略及控制参数值。

1.3 适应度地形分析技术

适应度地形(Fitness Landscape)是 Sewell Wright 于 1932 年提出的一个抽象的适合度值的二维等高线图, 在高维空间的进化过程直观图像^[38]。作为理解生物进化的一个抽象概念, 后来被用来分析问题的复杂程度以及智能优化算法的进化难易程度。研究学者已经提出很多适应度地形分析的技术, 通常使用随机游走的方法, 在问题空间中大量取样, 然后利用适应度地形分析技术对该问题进行特征提取, 以此体现问题适应度地形结构的平滑性、中立性和崎岖性。如果问题空间的适应度地形结构是光滑的地形, 这样的问题易于找到最优解, 这类地形具有平滑性; 适应度相等或几乎相等的区域, 这类地形具有中立性, 且具有欺骗性, 解决此类问题的优化算法易陷于局部最优; 在求解区域内呈现相对较大的起伏, 包含多个局部最优, 适应度地形结构呈陡升陡降之势, 这类地形具有崎岖性。

Li^[22]提出 DEET, 将进化划分为三种状态, 应用适应度-距离相关性分析技术划分当前种群进化状态, 然后依据进化状态, 自适应选择策略池, 为了避免算法陷入停滞, 提出了一种二级选择机制。Sallam^[40]提出了基于地形的自适应选择策略的差分进化算法 LSAOS-DE, 同时考虑了问题的适应度地形特征以及变异策略的历史经验信息, 对每一种变异策略进行了性能分析, 并在进化过程中自适应选

择合适的变异策略。Tan^[41-43]一些工作，采用适应度-距离相关性和信息熵方法构建问题的特征向量作为训练数据，再采用机器学习的一些方法：随机森林、k-最近邻算法、深度学习，离线训练多种变异策略与适应度地形特征之间的关系，根据模型预测适合解决该问题是哪种策略。Zhou^[21, 44]提出两项关于适应度地形分析技术的工作，首先第一个自适应邻域拓扑人工蜂群算法（简称ABC_ANT），采用适应度-距离相关性方法提取问题的结构特征，自适应选择邻域拓扑；第二个基于在线离散度量的人工蜂群算法（简称FLABC），采用在线离散度量分析问题的结构特征。

现有的一些适应度地形分析技术，需要进行大量的采样，如果应用于算法设计中，适应度地形分析技术需要消耗大量的计算资源，在有限的计算资源下，函数评估次数有限，相应的智能优化算法就缺少了计算资源，会导致算法求解性能不佳，因此传统的适应度地形分析技术适用于离线方式，难以结合到智能优化算法中。

1.4 本文主要工作及组织结构

本文针对现有的适应度地形分析技术应用于算法设计中，时间复杂度高，且求解性能不够高效。为此提出一种在线混合地形分析技术，在进化过程中实时分析问题的适应度地形结构特征，从问题的适应度地形结构特征角度考虑，设计算法，用于解决上述问题。如图 1-3 围绕在线混合地形分析技术，本文开展了三个主要工作。



图 1-3 组织结构

章节总体安排如下：

- (1) 绪论部分。首先介绍研究背景；其次介绍两种典型的智能优化算法，分别为人工蜂群算法和差分进化算法，简要说明算法流程，并对相关研究工作进行归纳总结；接着介绍了适应度地形分析技术；最后介绍了本文的主要工作及简要的组织结构。
- (2) 介绍几种相关的适应度地形分析技术，结合信息熵和信息地形两种分析技术的各自优点，本文提出了在线混合地形分析技术，详细说明了其计

算方法，并举例说明计算过程；最后章节总结。

- (3) 设计了基于在线混合地形分析技术的人工蜂群算法，首先给出引言；其次简要介绍 FLABC 算法；接着介绍本章提出的算法 FLABC-LM，并进行实验验证；最后章节总结。
- (4) 设计了基于在线混合地形分析技术的差分进化算法，首先给出引言；其次简要介绍 DEET 算法；接着介绍本章提出的算法 DEET-LM，并进行实验验证；最后章节总结。
- (5) 设计了基于在线混合地形分析技术的多策略人工蜂群算法，首先给出引言；其次简要介绍 MEABC 算法；接着介绍本章提出的算法 MEABC-LM，并进行实验验证；最后章节总结。
- (6) 设计了基于在线混合地形分析技术的集成式差分进化算法，首先给出引言；其次简要介绍 EDEV 算法；接着介绍本章提出的算法 EDEV-LM，并进行实验验证；最后章节总结。
- (7) 总结与展望。总结基于在线混合地形分析技术的三个工作，并指出未来工作的改进方向。

第二章 在线混合地形分析技术

2.1 代表性的适应度地形分析技术

简要介绍本文相关的适应度地形分析技术。适应度-距离相关性用来衡量问题的进化难易程度；离散度量用来分析问题特性是否存在单漏斗或者多漏斗结构；信息地形和信息熵用来分析问题的复杂程度。

2.1.1 适应度-距离相关性

最早由 Jones^[45]于 1995 年提出的适应度-距离相关性（简称 FDC）。其地形特征具有模态性，为了衡量遗传算法在组合优化问题上的问题进化难易程度。简单描述，在搜索空间中随机采样获得一组样本 S ， $distance$ 为样本 S 中最好的个体和所有个体之间的欧式距离， $fitness$ 样本 S 中所有个体和最优个体之间的适应度差值，计算 $distance$ 与 $fitness$ 之间的相关系数，以此量化问题的地形结构特征，相关系数的取值范围介于-1 到 1，如果相关系数愈趋近于 1，则表明地形是单模态结构，地形简单平滑，问题较容易求解；如果相关系数愈趋近于-1，则表明地形是多模态结构，地形崎岖，问题较难求解。下面是介绍相关系数 R 的计算公式：

$$R = \frac{C_{fd}}{\delta_{fitness}\delta_{distance}} \quad (2-1)$$

$$C_{fd} = \frac{1}{S} \cdot \sum_{i=1}^S (fitness_i - \overline{fitness}) \cdot (distance_i - \overline{distance}) \quad (2-2)$$

$$\delta_{fitness} = \sqrt{\frac{1}{S} \cdot \sum_{i=1}^S (fitness_i - \overline{fitness})^2} \quad (2-3)$$

$$\delta_{distance} = \sqrt{\frac{1}{S} \cdot \sum_{i=1}^S (distance_i - \overline{distance})^2} \quad (2-4)$$

其中 $\overline{distance}$ 和 $\overline{fitness}$ 分别表示距离和适应度的平均值， $\delta_{distance}$ 和 $\delta_{fitness}$ 分别是 $distance$ 和 $fitness$ 的标准差， C_{fd} 表示适应度和距离的协方差。

2.1.2 离散度量

Lunacek^[46]于 2006 年提出了一种新的度量标准来评估问题的结构特征，即离散度值（简称 DM）。该方法用于探究协方差矩阵自适应演化策略（CMA-ES）在某些多峰问题中表现不佳的原因，量化问题的地形结构特征是单漏斗或者多漏斗结构。该方法需对问题空间进行两次不同规模的随机抽样，并且第二次抽样的样本点要大于第一次样本点，然后选出每次抽样的样本适应度排名最好的前 N 个

体,并计算个体两两之间的欧氏距离,得到两个样本集离散值分别为 DM_1 、 DM_2 ;由于第二次抽样的样本点更多,样本点更加接近于全局最优,因此可以计算 DM_1 、 DM_2 之间的差值 DM_3 ,如果 DM_3 小于0,则表明该问题较为复杂,呈现多漏斗结构地形,这类问题往往较难解决,另外如果呈现为单漏斗地形,这类问题较容易求解。该方法通过两次采样,计算出两次抽样样本之间离散程度,分别得到两个离散度值 DM_1 、 DM_2 ,之后两个值作差,最终计算得到问题特征值 DM_3 。

2.1.3 信息地形

Borenstein^[47]于2005年提出了一种信息地形(简称IL)的度量方法,用来分析问题的复杂程度。简单概括为,在优化问题的搜索空间进行大量的抽样,样本容量为 M ,然后比较样本点两两之间适应度大小关系,以此构建 $M \times M$ 信息矩阵,信息矩阵元素 $m_{i,j}$ 计算公式(2-5)如下:

$$m_{i,j} = \begin{cases} 1 & \text{if } f(X_i) > f(X_j) \\ 0.5 & \text{if } f(X_i) = f(X_j) \\ 0 & \text{if } f(X_i) < f(X_j) \end{cases} \quad (2-5)$$

分别计算当前问题和参考问题的信息矩阵 M_1 和 M_2 ,“参考问题”一般是指地形结构简单的问题(已知全局最优),将当前问题信息矩阵和“参考问题”信息矩阵进行差分处理。差值愈大,表明优化问题的地形结构愈复杂;差值愈小,表明优化问题的地形结构愈简单。

2.1.4 信息熵

Malan^[48]于2009年提出基于信息熵(简称IE)的适应度地形度量技术,用来分析问题的崎岖程度。采用随机游走方式对问题空间进行大规模抽样,假设样本容量大小为 n ,获得适应度序列 $\{f_1, f_2, \dots, f_n\}$,其中 $\varepsilon^* = \max\{f_i - f_{i-1}\}$, ε 通常取9个值 $\{\varepsilon | \varepsilon = 0, \frac{\varepsilon^*}{128}, \frac{\varepsilon^*}{64}, \frac{\varepsilon^*}{32}, \frac{\varepsilon^*}{16}, \frac{\varepsilon^*}{8}, \frac{\varepsilon^*}{4}, \frac{\varepsilon^*}{2}, \varepsilon^*\}$,通过计算得到地形序列 $s(\varepsilon) = \{s_1, s_2, \dots, s_{n-1}\}$ 公式(2-6)如下:

$$s_i = \begin{cases} \bar{1} & \text{if } f_i - f_{i-1} < -\varepsilon \\ 0 & \text{if } |f_i - f_{i-1}| \leq \varepsilon \\ 1 & \text{if } f_i - f_{i-1} > \varepsilon \end{cases} \quad (2-6)$$

序列 $s(\varepsilon)$ 中连续符号值的组合含义如图2-1所示,可知不同符号组合6种。

统计 $s(\varepsilon)$ 序列中连续不同符号值的组合的频率 $P_{pq} = \frac{n_{pq}}{n}$,由此计算出 $H(\varepsilon)$:

$$H(\varepsilon) = -\sum_{p \neq q} P_{pq} \cdot \log_6 P_{pq} \quad (2-7)$$










符号值 pq	形状	类别
00		中性
01		粗糙
0 $\bar{1}$		粗糙
10		粗糙
11		光滑
1 $\bar{1}$		粗糙
$\bar{1}$ 0		粗糙
$\bar{1}$ 1		粗糙
$\bar{1}$ $\bar{1}$		光滑

图 2-1 连续符号值的组合含义^①

计算出 $H(\varepsilon)$ 的最大值，即信息熵为 $\max\{H(\varepsilon)\}$ 。信息熵量化问题的崎岖程度，连续不同符号值组合的频率越高，则表明地形结构越崎岖，连续不同符号值组合的频率越低，则表明地形结构越平滑。对于同一个函数，其适应度地形在不同的 ε 取值情况下，粗糙程度也不一定相同。

2.2 本文提出的在线混合地形分析技术

早期的适应度地形分析技术基本上都是采用离线方式量化地形结构特征。信息地形的优点是引入参考问题即理想地形具有平滑且简单的地形结构，但是需要问题搜索空间大量的采样，构建参考问题和当前问题的信息矩阵计算十分耗时，样本如果太少，存在样本偏差，提取问题特征不一定准确。信息熵的优点是引入地形序列，相对构建信息地形的信息矩阵，减少了计算量。另外这两种地形分析技术都能刻画问题的复杂程度。为了解决上述问题，本文提出一种在线混合地形分析技术，将信息地形和信息熵技术各自优点结合起来，引入信息熵的地形序列，以及结合信息地形中设置参考问题的方式，下面是详细的计算步骤：

- (1) 为方便集成到智能优化算法中，基于在线方式，以种群作为抽样样本，样本大小为 n ，得到一组解 $X = \{X_1, X_2, X_3, \dots, X_n\}$ ，同时计算在当前问题的适应度值，并从种群中找出最优个体；
- (2) 引入信息地形中设置参考问题，选定具有理想地形的参考问题，参考问题具有平滑性，参考问题选定带偏移的 Shift-Sphere 函数，根据当前种

^① 该图片来源于参考文献[59]：梁卓智. 基于适应度景观的差分进化算法策略研究[D]. 华南农业大学, 2019.

群中的最优个体的位置进行偏移。并在参考问题计算其适应度值。

- (3) 由于信息地形分析技术构建 $n*n$ 的信息矩阵，计算复杂度高，为了减少计算代价，引入信息熵的地形序列。计算出两个问题的适应度之后，通过比较相邻样本点之间适应度大小关系，通过公式(2-8)分别构建当前问题的地形序列 $S1 = \{S1_1, S1_2, S1_3, \dots, S1_{n-1}\}$ 和参考问题的地形序列 $S2 = \{S2_1, S2_2, S2_3, \dots, S2_{n-1}\}$ ，长度为 $n-1$ 。计算公式如下：

$$S_i = \begin{cases} \bar{1} & \text{if } f_i - f_{i-1} < 0 \\ 0 & \text{if } f_i - f_{i-1} = 0 \\ 1 & \text{if } f_i - f_{i-1} > 0 \end{cases} \quad (2-8)$$

- (4) 分别构建当前问题符号组合序列 $Sign1$ 和参考问题符号组合序列 $Sign2$ ，长度为 $n-2$ ，“ \vee ”表示两个字符进行拼接。目的是方便接下来统计符号组合不相同的频率，计算公式如下：

$$Sign(i) = S_i \vee S_{i+1} \quad (2-9)$$

- (5) 最后，统计两个符号组合序列不相同的次数 q ， q 越大，表明当前问题的地形与参考问题的平滑地形越不相似，说明当前问题地形越崎岖，以此量化地形结构特征的崎岖程度值，计算公式如下：

$$LM = \frac{q}{n-2} \quad (2-10)$$

LM 表示当前问题和参考问题地形结构特征的差异性， LM 的取值范围为 $[0,1]$ ，如果 LM 值越趋近于 0，则表明当前问题的地形结构越接近参考问题的地形结构，即地形更加平滑；如果 LM 值越趋近于 1，则表明当前问题的地形更加复杂崎岖。

算法 2-1: 在线混合地形分析技术的伪代码

```

01: 种群作为样本输入
02: 找出种群最优解，计算参考问题的适应度值
03: 用公式(2-8)分别计算当前问题和参考问题的地形序列S1和S2
04: 用公式(2-9)分别计算符号组合序列Sign1和Sign2，计数器 q=0
05:  for i = 1 : n - 2 do
06:     if Sign1(i)≠ Sign2(i)
07:         q =q + 1
08:     end if
09:  end for
10: 用公式(2-10)计算 LM 值
11: 输出 LM

```

为了更好的理解提出的在线混合地形分析技术,给出了在线混合地形分析技术算法 2-1 的伪代码。另外为更好地解释在线混合地形分析技术计算过程,引入一个问题实例,分析问题结构特征度量值 LM 是如何计算的。列举了 12 个候选解以及在两个问题上的函数值如表 2-1 所示。假设一个最小优化问题如公式(2-11):

$$f(x,y) = 2 \cdot (x^2 - y)^2 - (y - 1)^2 \quad (2-11)$$

表 2-1 12 个候选解以及在两个问题上的函数值

	(0,0)	(0,1)	(0,2)	(0,3)	(1,0)	(1,1)	(1,2)	(1,3)	(2,0)	(2,1)	(2,2)	(2,3)
当前问题	-1	2	7	14	1	0	1	4	31	18	7	-2
参考问题	13	8	5	4	10	5	2	1	9	4	1	0

- (1) 第一步,以 12 个候选解作为抽样样本,并计算在当前问题的适应度值,并找出样本中在当前问题中最优的解为(2,3);
- (2) 第二步,引入带偏移的 Shift-Sphere 函数,根据当前种群中的最优个体的位置进行偏移。其中候选解(2,3)可以看出是当前问题的最优解,因此参考问题为 Shift-Sphere 函数 $f(x,y) = (x - 2)^2 - (y - 3)^2$,并计算在当前问题的适应度值。
- (3) 第三步,首先根据公式 (2-8) 构建当前问题的地形序列 $S1=\{1, 1, 1, \bar{1}, \bar{1}, 1, 1, 1, \bar{1}, \bar{1}, \bar{1}\}$,以及参考问题的地形序列 $S2=\{\bar{1}, \bar{1}, \bar{1}, 1, \bar{1}, \bar{1}, \bar{1}, 1, \bar{1}, \bar{1}, \bar{1}\}$,长度为 $n-1$ 。
- (4) 第四步,根据公式 (2-9) 构建当前问题的符号组合序列 $Sign1$, $Sign1=\{11, 11, 1\bar{1}, \bar{1}\bar{1}, \bar{1}1, 11, 11, 1\bar{1}, \bar{1}\bar{1}, \bar{1}\bar{1}\}$,以及参考问题的地形序列 $Sign2=\{\bar{1}\bar{1}, \bar{1}\bar{1}, \bar{1}1, 1\bar{1}, \bar{1}\bar{1}, \bar{1}\bar{1}, \bar{1}1, 1\bar{1}, \bar{1}\bar{1}, \bar{1}\bar{1}\}$,长度为 $n-2$ 。
- (5) 第五步,统计两个符号组合序列不相同的次数 q ,根据公式(2-10)计算 LM 值。比较 $Sign1$ 和 $Sign2$ 符号组合出现不相同的次数 q 为 7,因此计算结果为 0.7,结果趋近于 1,该问题的地形结构特征较为复杂崎岖。

2.3 本章小结

简要介绍本文几种代表性的适应度地形分析技术,分别为适应度-距离相关性、离散度量、信息地形和信息熵。本文提出一种新的在线混合地形分析技术,将信息地形和信息熵技术各自优点结合起来,计算方式简单。

第三章 基于在线混合地形分析技术的人工蜂群算法

3.1 引言

对于复杂的优化问题，ABC 算法求解效果差强人意，原因在于其没有很好的平衡勘探能力以及开采能力，以往的 ABC 改进专注于算法自身角度，较少的研究工作从问题的适应度地形结构特征角度考虑。不同的解搜索方程处理相同的问题性能也不同，有些解搜索方程勘探能力较强，由于随机挑选个体引导算法搜索，确保种群的多样性，适合解决复杂的问题；另外还有些解搜索方程开采能力较强，利用较好的个体引导算法搜索，确保了收敛性，适合解决简单的问题。解搜索方程对于智能优化算法的搜索，起了关键作用，而如何有效地在进化过程中挑选出合适的策略，是目前急需解决的问题。基于此，引入适应度地形分析技术，分析问题的地形结构特征，算法根据特征自适应选择解搜索方程。Zhou 于 2022 年提出基于在线离散度量的人工蜂群算法^[21] (FLABC)，将在线离散度量首次应用于改进人工蜂群算法，但是在线离散度量存在时间复杂度高，算法求解性能不够高效。为此将在线混合地形分析技术替换在线离散度量分析技术，提出一种基于在线混合地形分析技术的人工蜂群算法 (FLABC-LM)。

3.2 FLABC 算法

FLABC 引入在线离散度量分析技术在进化过程分析问题结构特征，离散度量从问题漏斗性的角度考虑，漏斗类似盆地，通过计算离散度值来分析问题的整体结构，如果离散度值较高则呈现为多漏斗地形结构，意味着问题更复杂；如果离散度值呈现为单漏斗地形结构，意味着问题更简单。为了减少计算开销，FLABC 将离散度量改成基于在线方式，选择种群前 N 个最好的个体作为抽样样本，采样频率间隔代数为 20 代，计算不同进化代数的样本离散程度的差值即离散度量值，然后将地形分为单漏斗地形与多漏斗地形，在观察峰阶段，根据地形特征自适应选择解搜索方程，对于单漏斗结构采用开采能力强的解搜索方程，确保了收敛性；而对于多漏斗结构则采用勘探能力强的解搜索方程，确保了多样性。FLABC 从问题的地形结构角度考虑，在线离散度量分析技术能帮助刻画问题结构特征，并在进化过程中实时调整算法的搜索行为。

另外相对于基本 ABC，除了采用在线离散度量分析技术指导算法自适应选择策略，还有两个改进点，第一处改进点采用聚类划分技术，将种群动态划分为若干个子种群，有利于算法对不同区域进行高效搜索，确保了种群的多样性，促

进个体之间的信息交流；第二处改进点修改了侦查蜂阶段，设计新的邻域解搜索方程式(3-3)，替换随机初始化操作，以一定概率保留历史寻优经验，有利于种群收敛。图 3-1 给出了 FLABC 具体的伪代码。

Algorithm 3: Pseudocode of FLABC

```

1  According to Eq. (1), randomly generate  $SN$  food sources  $\{X_i | i = 1, 2, \dots, SN\}$  as an initial population;
2  Evaluate the fitness value of each food source and set  $FES = SN$ ;
3  Let  $G$  be the iteration index and set  $G = 0$ ;
4  while  $FES \leq MaxFES$  do
5      if  $G \% 20 = 0$  then
6          According to Algorithm 2, use the clustering partition technique to divide the population into dynamic multiple
           subpopulations;
7      end
      // ---Employed bee phase---
8      for  $i \leftarrow 1$  to  $SN$  do
9          Generate a new food source  $V_i$  for the food source  $X_i$  according to Eq. (7);
10         Evaluate the fitness value of  $V_i$  and set  $FES = FES + 1$ ;
11         if  $fit(X_i) < fit(V_i)$  then
12             Replace  $X_i$  with  $V_i$  and set  $trial_i = 0$ ;
13         else
14             Keep  $X_i$  to be the new food source and set  $trial_i = trial_i + 1$ ;
15         end
16     end
      // ---Onlooker bee phase---
17     According to Eq. (4), calculate the selection probability  $p_i$  of each food source;
18     Set  $t = 0$  and  $i = 0$ ;
19     while  $t < SN$  do
20         if  $rand(0, 1) < p_i$  then
21             if  $G < 20$  then
22                 Generate a new food source  $V_i$  for the selected food source  $X_i$  by randomly using Eqs. (8) or (9).
23             else
24                 if  $G \% 20 = 0$  then
25                     According to Algorithm 1, estimate the features of the local landscape by using the online FLA technique;
26                 end
27                 if Local landscape is smooth then
28                     Generate a new food source  $V_i$  for  $X_i$  according to Eq. (8);
29                 end
30                 if Local landscape is rugged then
31                     Generate a new food source  $V_i$  for  $X_i$  according to Eq. (9);
32                 end
33             end
34             Evaluate the fitness value of  $V_i$  and set  $FES = FES + 1$ ;
35             if  $fit(X_i) < fit(V_i)$  then
36                 Replace  $X_i$  with  $V_i$  and set  $trial_i = 0$ ;
37             else
38                 Keep  $X_i$  to be the new food source and set  $trial_i = trial_i + 1$ ;
39             end
40             Set  $t = t + 1$ ;
41         end
42     end
      Set  $i = (i + 1) \% SN$ ;
43 end
      // ---Scout bee phase---
44 if  $\max(trial_i) > limit$  then
45     According to Eq. (11), generate a new food source to replace the abandoned food source  $X_i$ ;
46     Set  $FES = FES + 1$  and  $trial_i = 0$ ;
47 end
48  $G = G + 1$ ;
49 end

```

图 3-1 FLABC 伪代码^①

下面介绍 FLABC 主要的解搜索方程：

^① 该算法伪代码来源于参考文献[21]: Zhou X, Song J, Wu S, et al. Artificial bee colony algorithm based on online fitness landscape analysis[J]. Information Sciences, 2023,619:603-629.

$$V_{i,j} = X_{lb,j} + \varphi_{i,j} \cdot [r \cdot X_{lb,j} + (1 - r) \cdot X_{i,j} - X_{k,j}] \quad (3-1)$$

$$V_{i,j} = \begin{cases} X_{lb,j} + \varphi_{i,j} \cdot [r \cdot X_{lb,j} + (1 - r) \cdot X_{gb,j} - X_{k,j}] & \text{if } landscape == 1 \\ X_{lb,j} + \varphi_{i,j} \cdot [r \cdot X_{lb,j} + (1 - r) \cdot X_{i,j} - X_{lk,j}] & \text{if } landscape == 2 \end{cases} \quad (3-2)$$

$$V_{i,j} = \begin{cases} r_1 X_{i,j} + r_2 X_{klb,j} + r_3 (X_{lk1,j} - X_{lk2,j}), & rand \leq CR \text{ or } j = j_{rand} \\ X_{i,j}, & otherwise \end{cases} \quad (3-3)$$

其中公式(3-1)用于雇佣蜂阶段，公式(3-2)用于观察蜂阶段，公式(3-3)用于侦查蜂阶段。 X_{lb} 表示包含 X_i 的子种群中最优的个体； X_k 是种群中随机挑选不同 X_i 的个体； X_{lk} 是从包含 X_i 的子种群中随机挑选不同 X_i 的个体； X_{gb} 是种群中最优的个体。 r 、 r_1 、 r_2 、 r_3 是随机数，取值范围为[0,1]， r_1 、 r_2 、 r_3 必须满足条件 $r_1+r_2+r_3=1$ ； X_{klb} 表示随机选择的子种群中最优的个体； X_{lk1} 和 X_{lk2} 表示随机选择的子种群中不同于 X_{klb} 两个个体； CR 控制子代个体继承父代个体经验信息的概率。

3.3 FLABC-LM 算法

FLABC-LM 与 FLABC 不同之处在于适应度地形分析技术，本文提出的在线混合地形分析技术替换了在线离散度量技术。为了方便说明 FLABC-LM，引入 FLABC 的伪代码如图 3-1 所示。算法 3-1 替换 FLABC 伪代码第 6 行。通过在线混合地形分析技术量化问题的地形结构特征，引入一个参数 u 控制地形结构划分，将地形结构划分为单漏斗结构或者多漏斗结构。如果地形结构特征为单漏斗，则采用开采能力强的解搜索方程；如果地形结构特征为多漏斗，则采用勘探能力较强的解搜索方程。其中参数 u 取值将在实验部分进行讨论。另外图 3-1 中第 27-32 行，根据算法 3-1 输出的 $landscape$ 的值，应用公式(3-2)自适应选择策略。

算法 3-1: FLABC-LM 算法的核心伪代码

```

01:   采用在线混合地形分析技术算法 2-1 计算 LM
02:   if LM < u
03:       表明为简单地形, landscape=1
04:   else
05:       表明为复杂地形, landscape=2
06:   输出 landscape
    
```

3.4 实验验证

为了验证 FLABC-LM 算法的性能，与 FLABC 进行对比，设计了四组实验：

(1) 参数敏感性分析。该实验针对关键的一个参数 u 进行参数敏感性分析。

运用在线混合地形分析技术量化问题的特征得到 LM ，再通过参数 u 控

制划分成两种地形特征。

- (2) FLABC-LM 与 FLABC 进行结果精度比较。目的是为了验证基于在线混合地形分析技术设计的算法，比基于在线离散度量设计的算法更加高效。
- (3) 策略有效性分析。该实验对于提出的在线混合地形分析技术进行策略有效性分析。
- (4) 运行时间的对比。该实验对于在线离散度量和本文提出的在线混合地形分析技术的时间复杂度的比较。

3.4.1 实验设置

在实验中，使用了两个测试集 CEC 2013^[49]以及 CEC 2014^[50]共包含了 58 个测试问题。CEC2013 测试集有 28 个基准测试函数，首先第一类包括 5 个单峰函数(F01-F05)、第二类包括了 15 个基本多峰函数(F06-F20)和第三类包括 8 个组合函数(F21-F28)，表 3-1 给出了 CEC2013 测试集的简介；CEC2014 测试集有 30 个基准测试函数，首先第一类包括 3 个单峰函数(F01-F03)、第二类包括 13 个简单的多峰函数(F04-F16)第三类包括 6 个混合函数(F17-F22)和第四类包括 8 个组合函数(F23-F30)，表 3-2 给出了 CEC2014 测试集。测试了两种维度： $D=30$ 和 $D=50$ 。 u 参数控制地形结构特征划分，设置为 0.5，其他实验参数设置与 FLABC 一致。最大评估次数 $MaxFes=10000 \cdot D$ ，算法独立运行 30 次，每次运行记录函数的误差值，并通过两种检验 Friedman^[51]和 Wilcoxon^[52]，其中 Wilcoxon 秩和检验分析算法之间在单个问题是否有显著性差异，显著性 α 设置为 0.05，如果显著性大于 α ，则表明无显著差异，用“ \approx ”表示；如果有显著性差异再比较平均标准误差大小，如果与对比的算法相比，结果好，则用“+”表示，否则用“-”表示。另外 Friedman 排名检验，用于比较多个算法在测试集的性能强弱，平均排名值越小，则表明算法性能越强。

表 3-1 CEC2013 测试集中 28 个基准函数简介

Func.	Name	Search range	Global minimum
F01	Sphere Function	[-100, 100]	-1400
F02	Rotated High Conditioned Elliptic Function	[-100, 100]	-1300
F03	Rotated Bent Cigar Function	[-100, 100]	-1200
F04	Rotated Discus Function	[-100, 100]	-1100
F05	Different Powers Function	[-100, 100]	-1000
F06	Rotated Rosenbrock's Function	[-100, 100]	-900
F07	Rotated Schaffers F7 Function	[-100, 100]	-800
F08	Rotated Ackley's Function	[-100, 100]	-700
F09	Rotated Weierstrass Function	[-100, 100]	-600
F10	Rotated Ghiewank's Function	[-100, 100]	-500

续表 3.1 CEC2013 测试集中 28 个基准函数简介

Func.	Name	Search range	Global minimum
F11	Rastrigin's Function	[-100, 100]	-400
F12	Rotated Rastrigin's Function	[-100, 100]	-300
F13	Non-Continuous Rotated Rastrigin's Function	[-100, 100]	-200
F14	Schwefel's Function	[-100, 100]	-100
F15	Rotated Schwefel's Function	[-100, 100]	100
F16	Rotated Katsuura Function	[-100, 100]	200
F17	Lunacek Bi_Rastrigin Function	[-100, 100]	300
F18	Rotated Lunacek Bi_Rastrigin Function	[-100, 100]	400
F19	Expanded Ghiewank's plus Rosenbrock's Function	[-100, 100]	500
F20	Expanded Scaffer's F6 Function	[-100, 100]	600
F21	Composition Function 1 (n=5, Rotated)	[-100, 100]	700
F22	Composition Function 2 (n=3, Unrotated)	[-100, 100]	800
F23	Composition Function 3 (n=3, Rotated)	[-100, 100]	900
F24	Composition Function 4 (n=3, Rotated)	[-100, 100]	1000
F25	Composition Function 5 (n=3, Rotated)	[-100, 100]	1100
F26	Composition Function 6 (n=5, Rotated)	[-100, 100]	1200
F27	Composition Function 7 (n=5, Rotated)	[-100, 100]	1300
F28	Composition Function 8 (n=5, Rotated)	[-100, 100]	1400

表 3-2 CEC2014 测试集中 30 个基准函数简介

Func.	Name	Search range	Global minimum
F01	Rotated High Conditioned Elliptic Function	[-100, 100]	100
F02	Rotated Bent Cigar Function	[-100, 100]	200
F03	Rotated Discus Function	[-100, 100]	300
F04	Shifted and Rotated Rosenbrock's Function	[-100, 100]	400
F05	Shifted and Rotated Ackley's Function	[-100, 100]	500
F06	Shifted and Rotated Weierstrass Function	[-100, 100]	600
F07	Shifted and Rotated Griewank's Function	[-100, 100]	700
F08	Shifted Rastrigin's Function	[-100, 100]	800
F09	Shifted and Rotated Rastrigin's Function	[-100, 100]	900
F10	Shifted Schwefel's Function	[-100, 100]	1000
F11	Shifted and Rotated Schwefel's Function	[-100, 100]	1100
F12	Shifted and Rotated Katsuura Function	[-100, 100]	1200
F13	Shifted and Rotated HappyCat Function	[-100, 100]	1300
F14	Shifted and Rotated HGBat Function	[-100, 100]	1400
F15	Shifted and Rotated Expanded Griewank's plus Rosenbrock's Function	[-100, 100]	1500
F16	Shifted and Rotated Expanded Scaffer's F6 Function	[-100, 100]	1600
F17	Hybrid Function 1 (N=3)	[-100, 100]	1700

续表 3.2 CEC2014 测试集中 30 个基准函数简介

Func.	Name	Search range	Global minimum
F18	Hybrid Function 2 (N=3)	[-100, 100]	1800
F19	Hybrid Function3 (N=4)	[-100, 100]	1900
F20	Hybrid Function3 (N=4)	[-100, 100]	2000
F21	Hybrid Function 5 (N=5)	[-100, 100]	2100
F22	Hybrid Function 6 (N=5)	[-100, 100]	2200
F23	Composition Function 1 (N=3)	[-100, 100]	2300
F24	Composition Function 2 (N=3)	[-100, 100]	2400
F25	Composition Function 3 (N=3)	[-100, 100]	2500
F26	Composition Function 4 (N=5)	[-100, 100]	2600
F27	Composition Function 5 (N=5)	[-100, 100]	2700
F28	Composition Function 6 (N=5)	[-100, 100]	2800
F29	Composition Function 7 (N=3)	[-100, 100]	2900
F30	Composition Function 8 (N=3)	[-100, 100]	3000

3.4.2 参数敏感性分析

通过在线混合地形分析技术将问题的地形结构特征进行量化得到 LM ，需要引入一个参数 u 将问题划分为两种类型地形特征，即简单地形和复杂地形，对于简单的地形，一般选用开采能力强的解搜索方程，确保了收敛性；而复杂的地形，一般采用勘探能力强的解搜索方程，确保了多样性。因此 u 的取值影响地形结构的划分，为了分析参数 u ，选取出几个具有代表性的值分别为 0.3、0.4、0.5、0.6、0.7，从 30D 的 CEC2013 上 28 个测试函数的实验数据看出，不同的 u 值对算法性能是有影响的。假如 u 值较小，算法较大概率选择勘探能力较强的解搜索方程，反之 u 值较大，算法较大概率选择开采能力较强的解搜索方程，因此合理设置 u 值，能较好地平衡算法的勘探和开采。从表 3-3 中可以看出， u 取 0.5 时，有 14 个函数取得较好的结果，算法的求解精度综合最好，并且其 Friedman 排名最高，因此 u 的值设置为 0.5。

表 3-3 FLABC-LM 参数 u 的敏感性分析在 CEC2013 比较

Function	$u=0.3(\text{Mean}\pm\text{Std})$	$u=0.4(\text{Mean}\pm\text{Std})$	$u=0.5(\text{Mean}\pm\text{Std})$	$u=0.6(\text{Mean}\pm\text{Std})$	$u=0.7(\text{Mean}\pm\text{Std})$
F01	2.27e-14±6.82e-14	2.27e-14±6.82e-14	4.55e-14±9.09e-14	5.31e-14±9.62e-14	3.03e-14±7.73e-14
F02	4.23e+06±6.95e+05	4.19e+06±1.11e+06	4.38e+06±1.15e+06	4.55e+06±7.77e+05	4.50e+06±1.19e+06
F03	8.49e+06±5.83e+06	1.07e+07±6.55e+06	7.05e+06±3.78e+06	1.07e+07±8.55e+06	9.93e+06±7.31e+06
F04	4.11e+04±3.41e+03	4.10e+04±3.66e+03	4.08e+04±3.82e+03	4.11e+04±3.59e+03	4.08e+04±4.09e+03
F05	4.96e-13±8.03e-14	4.81e-13±8.14e-14	4.93e-13±6.11e-14	4.66e-13±8.48e-14	4.81e-13±7.00e-14
F06	1.63e+01±2.65e-01	1.63e+01±4.75e-01	1.63e+01±4.20e-01	1.63e+01±6.62e-01	1.61e+01±1.56e+00

续表 3-3 FLABC-LM 参数 u 的敏感性分析在 CEC2013 比较

Function	$u=0.3(\text{Mean}\pm\text{Std})$	$u=0.4(\text{Mean}\pm\text{Std})$	$u=0.5(\text{Mean}\pm\text{Std})$	$u=0.6(\text{Mean}\pm\text{Std})$	$u=0.7(\text{Mean}\pm\text{Std})$
F07	4.54e+01±5.29e+00	4.43e+01±5.56e+00	4.26e+01±6.60e+00	4.31e+01±6.71e+00	4.30e+01±5.12e+00
F08	2.09e+01±5.17e-02	2.10e+01±4.63e-02	2.09e+01±4.43e-02	2.09e+01±6.10e-02	2.10e+01±4.07e-02
F09	2.07e+01±2.60e+00	2.05e+01±2.80e+00	2.03e+01±3.11e+00	2.02e+01±3.14e+00	2.10e+01±2.47e+00
F10	5.40e-01±2.58e-01	5.14e-01±2.17e-01	4.30e-01±2.18e-01	5.25e-01±2.37e-01	5.35e-01±2.94e-01
F11	0.00e+00±0.00e+00	0.00e+00±0.00e+00	0.00e+00±0.00e+00	0.00e+00±0.00e+00	1.89e-15±1.02e-14
F12	1.00e+02±1.17e+01	9.74e+01±1.29e+01	9.33e+01±9.87e+00	9.46e+01±9.34e+00	9.90e+01±9.08e+00
F13	1.51e+02±1.56e+01	1.51e+02±1.73e+01	1.48e+02±1.10e+01	1.46e+02±1.14e+01	1.52e+02±2.01e+01
F14	5.38e-02±3.19e-02	2.48e-02±1.94e-02	3.61e-02±2.23e-02	8.20e-02±2.22e-01	1.32e-01±2.92e-01
F15	2.92e+03±3.60e+02	3.11e+03±3.17e+02	3.04e+03±2.85e+02	2.92e+03±3.38e+02	3.02e+03±3.57e+02
F16	8.76e-01±1.47e-01	9.03e-01±1.51e-01	8.59e-01±1.73e-01	8.69e-01±1.41e-01	9.01e-01±1.60e-01
F17	3.04e+01±1.26e-03	3.04e+01±1.07e-14	3.04e+01±1.08e-03	3.04e+01±1.63e-03	3.04e+01±1.58e-03
F18	1.83e+02±1.45e+01	1.80e+02±1.69e+01	1.90e+02±1.24e+01	1.85e+02±1.91e+01	1.84e+02±1.58e+01
F19	6.69e-02±7.40e-02	6.08e-02±5.67e-02	6.39e-02±6.52e-02	7.23e-02±6.70e-02	6.36e-02±5.60e-02
F20	9.46e+00±4.48e-01	9.50e+00±3.92e-01	9.46e+00±3.29e-01	9.53e+00±3.30e-01	9.59e+00±2.90e-01
F21	2.20e+02±3.99e+01	2.27e+02±4.88e+01	2.36e+02±5.27e+01	2.31e+02±4.19e+01	2.44e+02±4.41e+01
F22	2.43e+01±2.92e+01	2.56e+01±3.27e+01	4.61e+01±4.67e+01	6.13e+01±4.90e+01	7.41e+01±4.66e+01
F23	4.06e+03±5.44e+02	4.03e+03±5.07e+02	4.00e+03±4.41e+02	3.96e+03±5.74e+02	4.15e+03±4.45e+02
F24	2.18e+02±1.90e+00	2.19e+02±2.63e+00	2.18e+02±2.79e+00	2.18e+02±2.99e+00	2.18e+02±2.21e+00
F25	2.29e+02±2.71e+01	2.36e+02±2.91e+01	2.38e+02±3.25e+01	2.35e+02±2.94e+01	2.40e+02±3.13e+01
F26	2.00e+02±3.13e-02	2.00e+02±3.14e-02	2.00e+02±2.93e-02	2.00e+02±3.21e-02	2.00e+02±3.20e-02
F27	4.49e+02±9.57e+01	4.68e+02±1.11e+02	4.62e+02±1.12e+02	4.61e+02±1.13e+02	4.72e+02±1.14e+02
F28	2.87e+02±4.99e+01	2.94e+02±3.28e+01	2.66e+02±7.04e+01	2.77e+02±5.94e+01	2.95e+02±2.14e+01
Friedman	2.8571	2.8214	2.4464	3.0179	3.8571

3.4.3 结果精度对比

如表 3-4 和 3-5 展示了 FLABC-LM 和 FLABC 在两套 CEC 测试集上，分别在 D=30 和 D=50 两个维度上的结果精度比对，明显可以看出 FLABC-LM 在大多数函数的求解精度要比 FLABC 好，根据 Friedman 排名结果，FLABC-LM 无论在两套测试函数且 30D 和 50D 的排名都要优于 FLABC。根据 Wilcoxon 检验，在两套测试函数不同维度的实验结果，FLABC-LM 在多峰函数显著优于 FLABC，而对于单峰函数以及复杂的混合函数少量好于 FLABC。总体上，FLABC-LM 在 58 个测试函数中，求解精度比 FLABC 高，FLABC-LM 比 FLABC 求解精度高的数量是 5-6 个，其它的测试函数求解精度相差不大。FLABC-LM 在 CEC2013 测试集，30D 和 50D 两个维度，其在函数 F10、F11、F14 上的结果精度都要比 FLABC 高。FLABC-LM 在 CEC2014 测试集，30D 和 50D 两个维度，其在函数 F8、F10、

F22 上的结果精度都要比 FLABC 高。相比于在线离散度量技术，在线混合地形分析技术能更好指导算法的搜索，提高了算法的性能。

表 3-4 FLABC-LM 在 CEC2013 测试集 30D 和 50D 结果精度比较

Function	30D		50D	
	FLABC(Mean±Std)	FLABC-LM(Mean±Std)	FLABC(Mean±Std)	FLABC-LM(Mean±Std)
F01	6.06e-14±1.01e-13≈	4.55e-14±9.09e-14	2.80e-13±9.62e-14+	2.27e-13±5.05e-29
F02	4.69e+06±8.01e+05≈	4.38e+06±1.15e+06	4.86e+06±1.19e+06≈	4.82e+06±9.68e+05
F03	1.00e+07±5.19e+06+	7.05e+06±3.78e+06	3.07e+07±1.36e+07≈	3.09e+07±1.79e+07
F04	4.15e+04±3.49e+03≈	4.08e+04±3.82e+03	6.27e+04±4.44e+03≈	6.19e+04±4.44e+03
F05	4.96e-13±1.16e-13≈	4.93e-13±6.11e-14	1.76e-12±1.63e-13≈	1.75e-12±1.62e-13
F06	1.63e+01±3.43e-01≈	1.63e+01±4.20e-01	4.40e+01±6.37e-01≈	4.37e+01±1.10e+00
F07	4.38e+01±5.50e+00≈	4.26e+01±6.60e+00	6.96e+01±5.24e+00≈	6.93e+01±4.68e+00
F08	2.09e+01±5.20e-02≈	2.09e+01±4.43e-02	2.11e+01±5.23e-02≈	2.11e+01±3.48e-02
F09	2.02e+01±2.47e+00≈	2.03e+01±3.11e+00	3.93e+01±2.97e+00≈	3.89e+01±2.68e+00
F10	6.06e-01±2.52e-01+	4.30e-01±2.18e-01	1.25e+00±1.05e-01+	1.18e+00±8.80e-02
F11	7.58e-15±1.93e-14+	0.00e+00±0.00e+00	6.44e-14±1.93e-14+	5.68e-14±3.79e-29
F12	9.71e+01±1.33e+01≈	9.33e+01±9.87e+00	2.17e+02±2.25e+01≈	2.14e+02±1.94e+01
F13	1.48e+02±1.84e+01≈	1.48e+02±1.10e+01	3.65e+02±3.55e+01≈	3.71e+02±3.54e+01
F14	1.11e-01±2.13e-01+	3.61e-02±2.23e-02	1.65e-01±4.13e-02+	1.23e-01±1.40e-01
F15	2.93e+03±3.33e+02≈	3.04e+03±2.85e+02	7.18e+03±4.90e+02≈	7.25e+03±5.03e+02
F16	8.93e-01±1.61e-01≈	8.59e-01±1.73e-01	1.16e+00±2.07e-01≈	1.20e+00±1.64e-01
F17	3.04e+01±5.59e-04≈	3.04e+01±1.08e-03	5.08e+01±2.01e-03+	5.08e+01±2.49e-04
F18	1.86e+02±1.62e+01≈	1.90e+02±1.24e+01	3.91e+02±3.32e+01≈	4.01e+02±2.77e+01
F19	7.91e-02±8.02e-02≈	6.39e-02±6.52e-02	1.94e-01±1.24e-01≈	1.75e-01±1.03e-01
F20	9.47e+00±3.51e-01≈	9.46e+00±3.29e-01	1.82e+01±2.90e-01≈	1.82e+01±4.24e-01
F21	2.48e+02±3.95e+01≈	2.36e+02±5.27e+01	2.86e+02±2.89e+01≈	2.72e+02±4.33e+01
F22	4.53e+01±4.53e+01≈	4.61e+01±4.67e+01	1.12e+01±1.98e+00≈	1.06e+01±1.80e+00
F23	4.07e+03±5.05e+02≈	4.00e+03±4.41e+02	1.05e+04±7.49e+02≈	1.04e+04±8.88e+02
F24	2.18e+02±2.03e+00≈	2.18e+02±2.79e+00	2.79e+02±5.36e+00≈	2.77e+02±5.04e+00
F25	2.38e+02±2.72e+01≈	2.38e+02±3.25e+01	3.77e+02±1.16e+01≈	3.80e+02±1.02e+01
F26	2.00e+02±2.54e-02≈	2.00e+02±2.93e-02	2.01e+02±1.12e-01≈	2.01e+02±1.19e-01
F27	5.08e+02±1.35e+02+	4.62e+02±1.12e+02	1.13e+03±3.66e+02≈	1.13e+03±3.25e+02
F28	2.81e+02±4.83e+01≈	2.66e+02±7.04e+01	4.00e+02±0.00e+00≈	4.00e+02±0.00e+00
+/-/≈	5/0/23		5/0/23	
Friedman	1.8214	1.1786	1.7321	1.2679

表 3-5 FLABC-LM 在 CEC2014 测试集 30D 和 50D 结果精度比较

Function	30D		50D	
	FLABC(Mean±Std)	FLABC-LM(Mean±Std)	FLABC(Mean±Std)	FLABC-LM(Mean±Std)
F01	3.90e+06±1.43e+06≈	3.66e+06±1.77e+06	3.91e+06±8.95e+05≈	3.73e+06±7.60e+05
F02	1.54e+02±3.47e+02≈	1.53e+02±2.43e+02	1.88e+03±1.77e+03+	5.94e+02±4.71e+02
F03	6.59e+02±4.80e+02≈	6.11e+02±5.80e+02	2.46e+03±9.87e+02≈	2.29e+03±8.78e+02
F04	1.06e+01±2.36e+01≈	5.31e+00±1.68e+01	6.19e+01±3.32e+01≈	5.69e+01±3.28e+01
F05	2.04e+01±5.49e-02≈	2.03e+01±5.05e-02	2.05e+01±8.03e-02+	2.04e+01±7.78e-02
F06	2.18e+00±1.22e+00≈	2.43e+00±1.26e+00	3.01e+00±1.25e+00≈	3.20e+00±1.13e+00
F07	8.56e-09±2.32e-08≈	7.61e-09±1.95e-08	2.09e-06±3.39e-06≈	1.18e-06±1.90e-06
F08	1.52e-14±3.86e-14+	0.00e+00±0.00e+00	1.33e-13±4.24e-14+	1.14e-13±7.57e-29
F09	2.15e+01±2.83e+00≈	2.10e+01±3.18e+00	4.93e+01±6.81e+00≈	4.91e+01±6.49e+00
F10	7.89e-02±3.99e-02+	3.71e-02±2.65e-02	1.93e-01±1.63e-01+	7.77e-02±3.63e-02
F11	9.25e+02±3.25e+02≈	8.97e+02±3.02e+02	2.59e+03±4.75e+02≈	2.53e+03±4.10e+02
F12	4.36e-01±6.58e-02+	3.77e-01±7.27e-02	4.18e-01±8.04e-02≈	4.10e-01±7.70e-02
F13	1.97e-01±1.41e-02≈	2.00e-01±2.17e-02	3.66e-01±3.53e-02≈	3.66e-01±2.31e-02
F14	1.23e-01±8.45e-03+	1.18e-01±1.13e-02	2.01e-01±1.02e-02≈	1.98e-01±1.42e-02
F15	4.64e+00±6.04e-01≈	4.34e+00±8.41e-01	1.14e+01±1.33e+00≈	1.16e+01±1.32e+00
F16	8.20e+00±5.01e-01≈	8.13e+00±3.93e-01	1.59e+01±3.73e-01≈	1.58e+01±4.55e-01
F17	1.14e+06±3.42e+05≈	1.12e+06±3.91e+05	1.03e+06±2.76e+05≈	1.02e+06±2.67e+05
F18	5.94e+02±8.96e+02≈	4.42e+02±5.18e+02	1.11e+03±6.87e+02≈	1.05e+03±7.36e+02
F19	5.35e+00±6.41e-01≈	5.38e+00±6.16e-01	1.29e+01±5.35e+00≈	1.33e+01±8.15e+00
F20	5.35e+03±2.30e+03≈	5.38e+03±1.71e+03	8.32e+03±1.31e+03≈	8.01e+03±1.81e+03
F21	1.46e+05±5.24e+04≈	1.55e+05±5.81e+04	7.18e+05±2.25e+05≈	6.11e+05±2.28e+05
F22	1.71e+02±4.51e+01+	1.54e+02±4.34e+01	5.06e+02±1.19e+02+	4.01e+02±1.07e+02
F23	3.15e+02±7.12e-03≈	3.15e+02±1.96e-02	3.44e+02±2.60e-02≈	3.44e+02±4.94e-02
F24	2.00e+02±2.86e-02≈	2.00e+02±5.60e-02	2.00e+02±1.22e-01≈	2.00e+02±9.38e-02
F25	2.05e+02±3.92e-01≈	2.06e+02±4.54e-01	2.12e+02±8.26e-01≈	2.12e+02±7.72e-01
F26	1.00e+02±2.21e-02≈	1.00e+02±2.64e-02	1.00e+02±1.59e-02≈	1.00e+02±2.11e-02
F27	3.22e+02±2.63e+01+	3.11e+02±1.85e+01	3.84e+02±4.83e+01≈	3.87e+02±5.73e+01
F28	8.14e+02±1.84e+01≈	8.09e+02±3.08e+01	1.18e+03±4.20e+01≈	1.18e+03±3.63e+01
F29	1.23e+03±1.44e+02≈	1.27e+03±1.52e+02	1.55e+03±2.70e+02≈	1.54e+03±2.19e+02
F30	1.75e+03±4.01e+02≈	1.92e+03±4.89e+02	8.62e+03±3.16e+02≈	8.66e+03±2.35e+02
+/-/≈	6/0/24		5/0/25	
Friedman	1.6333	1.3667	1.6667	1.3333

3.4.4 策略有效性验证

在线混合地形分析技术作为 FLABC-LM 算法的重要部分，在进化过程中实

时量化问题的地形结构特征，再利用该特征指导算法的搜索。在本节中，设计了两个新的算法变体分别为，第一，将信息地形分析技术替换在线离散度量技术设计出一个新的算法简称 FLABC-IL；第二，将信息熵分析技术替换在线离散度量技术设计出一个新的算法简称 FLABC-IE，同样是基于在线方式，以种群作为采样样本，以此量化当前问题的地形特征，算法参数与 FLABC-LM 保持一致，均在 30D 的 CEC2013 上进行测试。引入 FLABC-IL 和 FLABC-IE 这两个变体，目的是为了验证提出的在线混合地形分析技术有效性。

从表 3-6 给出的实验结果所示，首先根据 Friedman 排名检验结果，Friedman 的值越小，则表明算法的性能最好，提出的在线混合地形分析技术要比其他两种适应度地形分析技术性能更好；另外使用 Wilcoxon 检验，对比结果可以看出，FLABC-LM 比 FLABC-IL 在函数 F10、F12、F14、F22 有更好的结果，其他的函数结果则是相似；FLABC-LM 比 FLABC-IE 在函数 F1、F5、F11、F14、F16 有更好的结果，在 F18 函数 FLABC-LM 结果更差，其他的函数结果则是相似。从求解精度结果上看，FLABC-LM 有 17 个函数结果精度较好，领先其他两个算法 FLABC-IL 和 FLABC-IE。

表 3-6 FLABC-LM 策略有效性分析在 CEC2013 比较

Function	FLABC-IL(Mean±Std)	FLABC-IE(Mean±Std)	FLABC-LM(Mean±Std)
F01	1.52e-14±5.67e-14 ≈	1.59e-13±1.04e-13+	4.55e-14±9.09e-14
F02	4.52e+06±8.60e+05≈	4.03e+06±7.70e+05 ≈	4.38e+06±1.15e+06
F03	1.05e+07±7.94e+06≈	9.63e+06±6.61e+06≈	7.05e+06±3.78e+06
F04	4.08e+04±3.46e+03≈	4.26e+04±4.36e+03≈	4.08e+04±3.82e+03
F05	4.89e-13±7.28e-14 ≈	5.61e-13±7.73e-14+	4.93e-13±6.11e-14
F06	1.63e+01±8.05e-01≈	1.62e+01±6.01e-01 ≈	1.63e+01±4.20e-01
F07	4.29e+01±4.45e+00≈	4.38e+01±4.45e+00≈	4.26e+01±6.60e+00
F08	2.10e+01±4.76e-02≈	2.10e+01±3.79e-02≈	2.09e+01±4.43e-02
F09	2.11e+01±3.18e+00≈	2.10e+01±3.08e+00≈	2.03e+01±3.11e+00
F10	6.36e-01±4.70e-01+	4.81e-01±2.56e-01≈	4.30e-01±2.18e-01
F11	0.00e+00±0.00e+00 ≈	5.12e-14±1.71e-14+	0.00e+00±0.00e+00
F12	9.82e+01±1.11e+01+	9.43e+01±1.16e+01≈	9.33e+01±9.87e+00
F13	1.52e+02±1.67e+01≈	1.49e+02±1.44e+01≈	1.48e+02±1.10e+01
F14	8.70e-02±1.64e-01+	5.79e-02±4.03e-02+	3.61e-02±2.23e-02
F15	2.96e+03±3.18e+02 ≈	2.94e+03±3.58e+02≈	3.04e+03±2.85e+02
F16	8.89e-01±1.73e-01≈	9.50e-01±1.49e-01+	8.59e-01±1.73e-01
F17	3.04e+01±9.68e-03 ≈	3.04e+01±2.49e-04 ≈	3.04e+01±1.08e-03
F18	1.85e+02±1.34e+01≈	1.81e+02±1.63e+01 -	1.90e+02±1.24e+01
F19	6.69e-02±6.53e-02≈	8.32e-02±1.26e-01≈	6.39e-02±6.52e-02
F20	9.54e+00±3.93e-01≈	9.29e+00±1.45e+00 ≈	9.46e+00±3.29e-01

续表 3-6 FLABC-LM 策略有效性分析在 CEC2013 比较

Function	FLABC-IL(Mean±Std)	FLABC-IE(Mean±Std)	FLABC-LM(Mean±Std)
F21	2.50e+02±4.35e+01≈	2.24e+02±4.22e+01≈	2.36e+02±5.27e+01
F22	8.06e+01±4.38e+01+	1.75e+01±1.73e+01≈	4.61e+01±4.67e+01
F23	3.94e+03±4.45e+02≈	4.14e+03±4.39e+02≈	4.00e+03±4.41e+02
F24	2.18e+02±2.72e+00≈	2.18e+02±2.40e+00≈	2.18e+02±2.79e+00
F25	2.31e+02±2.55e+01≈	2.34e+02±2.84e+01≈	2.38e+02±3.25e+01
F26	2.00e+02±2.32e-02≈	2.00e+02±3.28e-02≈	2.00e+02±2.93e-02
F27	5.22e+02±1.38e+02≈	4.33e+02±8.07e+01≈	4.62e+02±1.12e+02
F28	2.81e+02±5.00e+01≈	2.76e+02±6.17e+01≈	2.66e+02±7.04e+01
+/-/≈	4/0/24	5/1/22	
Friedman	2.3393	2.0357	1.6250

3.4.5 运行时间对比

本小节验证两种适应度地形分析技术的时间复杂度，即在线离散度量和在线混合地形分析技术的运行时间的比较。为了直观表示，将运行时间进行量化处理，绘制成雷达图，覆盖区域越大，则表明该算法在时间上更耗时。从图 3-2 中可以看出，在单峰函数和多峰函数问题中，在线离散度量在大多数函数比在线混合地形分析技术更加耗时，而在一些混合函数，两种地形分析技术运行时间相差不大，综合来看，本文提出适应度地形分析技术时间复杂度更低。

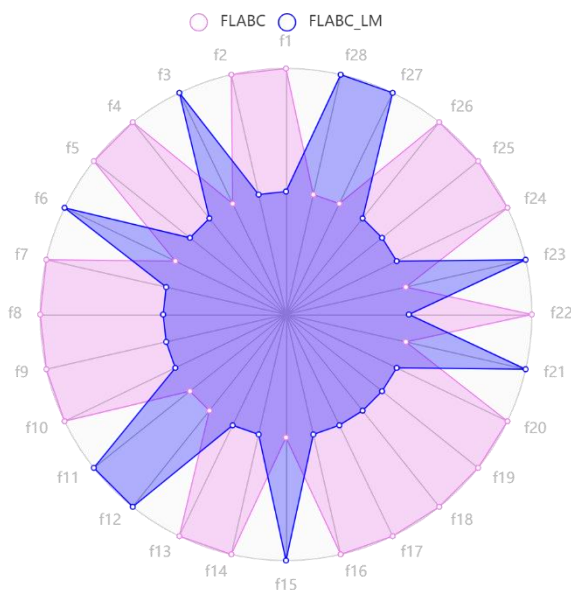


图 3-2 FLABC-LM 与 FLABC 的运行时间对比

3.5 本章小结

本文提出的在线混合地形分析技术比在线离散度量更加高效,且时间复杂度更低。本章的重点在于从地形结构特征的角度出发,根据特征自适应选择策略,设计了一种基于在线混合地形分析技术的人工蜂群算法。为了验证在线混合地形分析技术的有效性,设置了四组实验,并在两套测试集 CEC2013 和 CEC2014 的 30D 和 50D 两种维度下进行大量实验验证。实验证明,本文提出的技术更加高效,且运行时间更短。

第四章 基于在线混合地形分析技术的差分进化算法

4.1 引言

差分进化算法的性能在很大程度上取决于变异策略，因此如何灵活运用变异策略成为重要的研究热点。近年来，开始有少量的研究工作从问题的地形结构特征角度设计算法。Li 于 2019 年提出的进化状态估计和二级选择机制的差分进化算法 (DEET)，在进化过程中采用适应度-距离相关性分析技术，实时分析种群的进化状态，算法再根据进化状态自适应调整变异策略。但是根据适应度-距离相关性计算方式，具体描述同 2.1.1 小节，计算步骤繁多，时间复杂度高。为了进一步验证在线混合地形分析技术更加高效，且计算时间复杂度更低，因此设计了一种基于在线混合地形分析技术的差分进化算法 (DEET-LM)，在 DEET 的基础上，引入了在线混合地形分析技术来简化计算并降低了时间复杂度，替换了适应度-距离相关性分析技术。

4.2 DEET 算法

DEET 采用适应度-距离相关性作为进化状态估计方法，该方法计算种群中个体与最优个体的欧式距离和个体适应度之间的相关系数来量化问题的难易程度。相关系数的取值范围为 $[-1,1]$ ，相关系数越趋近于 1，则说明适应度地形结构越简单；相关系数越趋近于-1，则说明适应度地形结构越复杂崎岖。每一代都计算相关系数，由参数 u_{fac} 将地形结构分为三类 (DEET 设置 u_{fac} 为 0.85)，分别为简单地形、综合地形以及复杂地形。对于简单地形应选择开采状态的变异策略，增强种群的收敛能力，综合地形应选择平衡状态的变异策略，兼顾勘探和开采，此外对于复杂地形应选择勘探状态，确保种群的多样性。随后为每个个体自适应地从对应的候选池中选择一个变异策略来生成子代个体。图 4-1 给出了 DEET 具体的伪代码。

根据适应度-距离相关性分析技术量化问题特征得到 R 值，并设定阈值 u_{fac} 来划分进化状态，演化过程分为如下三种状态，具体公式如下：

$$S = \begin{cases} S_1: \text{开采状态} & \text{if } R > u_{fac} \\ S_2: \text{平衡状态} & \text{if } 1 - u_{fac} \leq R \leq u_{fac} \\ S_3: \text{勘探状态} & \text{if } R < 1 - u_{fac} \end{cases} \quad (4-1)$$

Algorithm 2: DEET

```

01: Set the population size  $NP$ ; Generate and evaluate the population  $\mathbf{P}^G$ ; set  $G=0$ ,
    Set control parameter  $\mu, Q, p, c_1$ ; Initialize  $p_{i,1} = p_{i,2} = 0.5, (i = 1, 2, 3)$ ;
    Initialize  $q_i = 0, (i = 1, 2, \dots, NP), \mu_F = 0.5, \mu_{CR} = 0.5, \text{Archive } A = [], FES = NP$ 
02: while the terminal condition is not satisfied
03:     Set  $S_F = [] S_{CR} = []$ , trial population  $\mathbf{U}^G = []$ 
04:     Calculate correlation coefficient  $c$  as Eq. (10)
05:     Determine the current evolutionary state  $S$  as Eq. (17)
06:     for  $i=1: NP$ 
07:         Generate  $F_i = \text{randc}(\mu_F, 0.1) CR_i = \text{randc}(\mu_{CR}, 0.1)$ 
08:         Select a mutation operator from the state ( $S$ ) in the operator pool based on  $p_{i,1}$  and  $p_{i,2}$ 
09:         Generate mutation vector  $V_i^G$  based on the selected mutation operator
10:         Generate trial vector  $U_i^G$  as Eq. (7).  $U_i^G \rightarrow \mathbf{U}^G$ 
11:     end
12:     Set  $I_{o1} = I_{o2} = 0$ 
13:     for  $i=1: NP$ 
14:         if  $f(U_i^G) < f(X_i^G)$ 
15:              $I_{ok} = I_{ok} + f(X_i^G) - f(U_i^G), k = 1$  or  $2$  based on which operator is used to generate  $V_i^G$ .
16:              $F_i \rightarrow S_F, CR_i \rightarrow S_{CR}$ 
17:         end if
18:         if  $f(U_i^G) \leq f(X_i^G)$ 
19:             Exchange  $X_i^G$  and  $U_i^G, q_i = 0$ 
20:             if the size of  $A$  is less than  $NP$ 
21:                  $X_i^G \rightarrow A$ 
22:             else
23:                 Randomly select an individual  $X_r$  from archive  $A$ 
24:                 Replace  $X_r$  with  $X_i^G$  if  $X_i^G$  is not worse than  $X_r$ 
25:             end if
26:         else
27:              $q_i = q_i + 1$ 
28:         end if
29:     end for
30:     ASO( $\mathbf{P}^G, \mathbf{U}^G, q, Q$ )
31:     Update corresponding  $p_{i,1}$  and  $p_{i,2}$  for the operator pool associated with state  $S$  as Eq. (24)
32:      $\mu_F = (1 - c_1) \cdot \mu_F + c_1 \cdot \text{mean}_i(S_F)$ 
33:      $\mu_{CR} = (1 - c_1) \cdot \mu_{CR} + c_1 \cdot \text{mean}_A(S_{CR})$ 
34:      $G = G + 1; FES = FES + NP$ 
35: end while

```

图 4-1 DEET 伪代码^①

其中，问题地形结构特征值 R 的大小决定了优化问题的地形类型。如果 R 大于阈值 u_{fac} ，表明当前问题归属于简单地形，因此采用开采能力较强的 S_1 策略池；如果 R 小于阈值 $1 - u_{fac}$ ，那么当前问题归属于复杂地形，因此采用勘探能力较强的 S_3 策略池；否则当前的问题归属于综合地形，因此采用兼顾勘探和开采的 S_2 策略池。

(1) S_1 :

$$V_i = X_i + F \cdot (X_{\text{best}} - X_i) + F \cdot (X_{r1} - X_{r\text{Archive}}) \quad (4-2)$$

$$V_i = X_{r1} + F \cdot (X_{\text{best}} - X_{r1}) + F \cdot (X_{r2} - X_{r\text{Archive}}) \quad (4-3)$$

^① 该算法伪代码来源于参考文献[22]: Li Y, Li G. Differential evolutionary algorithm with an evolutionary state estimation method and a two-level selection mechanism[J]. Soft Computing, 2020,24(15):11561-11581.

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/148133125135006023>