

多模态多目标优化综述

岳彩通¹, 梁静^{1†}, 瞿博阳², 于坤杰¹, 王艳丽¹, 胡毅¹

(1. 郑州大学电气工程学院, 郑州 450001; 2. 中原工学院电子信息学院, 郑州 450007)

摘要: 随着工业生产和日常生活需求的多样化, 单个解决方案已经无法满足生产生活的需求. 多模态优化可以为决策者提供多个可行方案, 但是早期对多模态优化的研究局限在单目标优化中. 在多目标优化中, 也存在多模态优化问题, 这些问题存在多个全局或局部帕累托最优解集, 找到这些最优解集具有重大的理论和实际意义. 本文首先介绍了多模态多目标优化问题的特点和求解难点; 然后综述了求解此类问题的主要方法, 总结了这些方法的优缺点; 随后介绍了常用的多模态多目标优化标准测试函数集和性能评价指标; 最后给出了多模态多目标优化的应用领域及未来研究方向.

关键词: 多模态多目标优化; 多模态优化; 测试函数; 全局优化

中图分类号: TP273 文献标志码: A

DOI: 10.13195/j.kzyjc.2020.1509

A Survey on Multimodal Multiobjective Optimization

YUE Cai-tong¹, LIANG Jing^{1†}, QU Bo-yang², YU Kun-jie¹, WANG Yan-li¹, HU Yi¹

(1. School of Electrical Engineering, Zhengzhou University, Zhengzhou 450001, China; 2. School of Electric & Information Engineering, Zhongyuan University of Technology, Zhengzhou 450007, China)

Abstract: As the requirements in our daily lives and industrial production become diverse, a single solution cannot meet their demands anymore. Multimodal optimization can provide multiple feasible solutions. However, most of the previous researches on multimodal optimization focus on single-objective optimization. In fact, there are multimodal problems in multiobjective optimization. These problems have multiple local or global Pareto sets and it is of great theoretical and practical significance to find these Pareto sets. The major contents of this paper are summarized as follows. First, it introduces the features and challenges of multimodal multiobjective problems. Second, a survey on existing multimodal multiobjective optimization algorithms is given. Third, multimodal multiobjective benchmark problems and performance indicators are introduced. At last, the applications and future research works are given.

Keywords: multimodal multiobjective optimization; multimodal optimization; test function; global optimization

0 引言

实际应用中存在大量多模态优化问题, 传统意义上的多模态优化是指多模态单目标优化(即多峰优化)^[1-5]. 对多模态单目标优化的研究由来已久, 解决此类问题最常用的方法是小生境策略^[3, 6], 该方法使得进化在局部范围内进行, 防止算法过早收敛到某一个区域.

许多优化问题的目标不止一个, 而是有两个或两个以上的待优化目标, 且改进其中一个目标可能会造成其他目标的恶化, 这些问题被称作多目标优化问题^[7-12]. 一般情况下, 不可能同时找到所有目标的最优值. 常用于解决多目标优化问题的方法是找到不同目标之间的最优均衡解集. 这个解集分布在

帕累托前沿上, 为决策者提供更多选择. 由于解的优劣都根据目标值(即在目标空间)进行评估, 因此决策空间的分布特性没有受到太多关注. 早期研究的多目标优化问题都被默认为单模态多目标优化问题, 即问题只有一个帕累托最优解集.

事实上, 多目标优化中也存在多模态情形, 即在某些多目标优化问题中存在多个全局或局部帕累托最优解集的情况, 这些存在多模态情形的多目标优化问题被称为多模态多目标优化问题^[13-16]. 图 1 给出了一个简单的多模态多目标优化实例^[17]. 该实例是特征选择优化问题^[18, 19], 此问题有两个待优化目标: (1) 选择特征的个数; (2) 分类错误率. 图中的虚线表示此问题的帕累托前沿, 如图所示, 当选择一个

收稿日期: 2020-11-01; 修回日期: 2021-01-21.

基金项目: 国家自然科学基金项目(61922072, 61976237, 61876169, 61673404 and 61806179)

[†]通讯作者. E-mail: liangjing@zzu.edu.cn.

特征时,有三个不同的选择方案: $\{F_1\}$, $\{F_2\}$ 或 $\{F_4\}$. 这三种方案的两个目标值都相同,即选择 F_1 , F_2 或 F_4 , 选择特征的个数都是 1, 错误率都是 0.8. 如果选择两个特征, 那么选择 $\{F_2, F_3\}$ 或 $\{F_1, F_4\}$ 的错误率都是 0.4. 该特征选择问题的帕累托最优解集有多种组合(只考虑选择特征个数为 1 或者 2 的组合就有 $2 \times 3 = 6$ 种), 也即它有多组帕累托前沿, 因此该问题属于多模态多目标优化问题. 在该问题中, 一组帕累托最优解集不能很好地满足决策者的需求. 因为提取 F_1 和 F_4 需要花费的时间、人力、物力等资源较多, 而 F_2 和 F_3 都比较容易提取, 如果选择一个特征时算法找到的最优特征是 F_1 , 选择两个特征时算法找到的最优特征组合是 $\{F_1, F_4\}$, 那么决策者不得不消耗大量的资源去提取 F_1 和 F_4 . 如果选择一个特征时算法找到的最优特征是 F_1, F_2 或 F_4 , 选择两个特征时算法找到的最优特征组合是 $\{F_1, F_4\}$ 或 $\{F_2, F_3\}$ 这两种特征组合, 那么决策者可以选择 F_2 (需要一个特征时), $\{F_2, F_3\}$ (需要两个特征时), 这既能满足较低的错误率, 又能节约资源. 因为现有多模态多目标优化方法是实数空间的连续优化方法, 因此本文只对连续多模态多目标优化方法进行了综述.

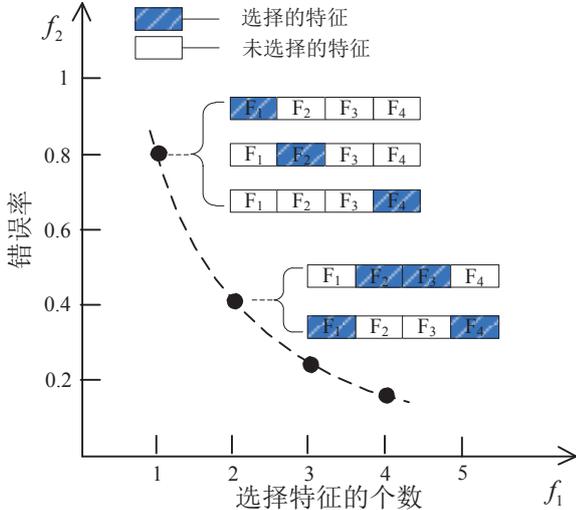


图1 多模态多目标特征选择

多模态多目标优化方法需要保留多个帕累托解集, 其意义^[3]在于: (1) 多个解集可以满足不同决策者的需求; (2) 多个解集可以揭示问题的潜在特性; (3) 快速由一个解集转化到另一个解集可以帮助解决动态优化问题; (4) 提供多个解集可以提高找到鲁棒解的可能性.

传统多目标优化算法难以找到多个帕累托最优解集, 因为传统多目标优化算法极少利用解集在决策空间的分布特性, 它们侧重于提高目标空间的多

样性、延展性和收敛性. 为了找到多个帕累托最优解, 需要设计特殊的搜索机制和环境选择策略.

本文对解决多模态多目标优化问题的相关文献进行了分析总结, 介绍了多模态多目标优化问题的特点和难点, 分析了现有的解决方法并总结了它们各自的优缺点, 介绍了关于多模态多目标优化的测试问题和评价指标, 希望能给该领域的研究人员提供参考.

1 相关定义

不失一般性, 最小化的多目标优化问题定义如下:

$$\begin{aligned} \text{Min } \vec{f}(\vec{x}) &= [f_1(\vec{x}), f_2(\vec{x}), \dots, f_m(\vec{x})] \\ \text{s.t.} & \\ g_i(\vec{x}) &\leq 0 \quad i = 1, 2, \dots, k \\ h_j(\vec{x}) &= 0 \quad j = 1, 2, \dots, p \end{aligned} \quad (1)$$

其中 m 表示待优化目标的数目, $\vec{x} = (x_1, x_2, \dots, x_n)$ 是 n 维决策变量, $g_i(\vec{x}) \leq 0$ ($i = 1, 2, \dots, k$) 表示 k 个不等式约束, $h_j(\vec{x}) = 0$ ($j = 1, 2, \dots, p$) 表示 p 个等式约束, 同时满足上述等式和不等式约束的空间 \mathbb{R}^n 称为决策空间; 决策空间通过目标函数映射出的空间 \mathbb{R}^m 称为目标空间.

定义 1: 支配关系^[20]

给定一个多目标优化问题 $\text{Min } \vec{f}(\vec{x}) = [f_1(\vec{x}), f_2(\vec{x}), \dots, f_m(\vec{x})]$, 其中 m 表示目标的个数. 假设 \vec{x}_1 和 \vec{x}_2 是该优化问题的两个可行解, 当且仅当 $\forall i = 1, 2, \dots, m, f_i(\vec{x}_1) \leq f_i(\vec{x}_2) \wedge \exists j \in [1, m], f_j(\vec{x}_1) < f_j(\vec{x}_2)$ 时称 \vec{x}_1 支配 \vec{x}_2 .

如果某个解不被其他任何解支配, 那么这个解被称为非支配解. 所有的非支配解组成帕累托最优解集 (Pareto optimal Set, PS); PS 映射到目标空间的集合被称为帕累托前沿 (Pareto Front, PF).

定义 2: 多模态多目标优化^[21, 22]

如果一个多目标优化问题满足以下两个条件之一, 那么该问题被称为多模态多目标优化问题:

- (1) 该问题至少有一个局部帕累托最优解;
- (2) 该问题至少有两个等效全局帕累托最优解, 它们对应 PF 上同一个点.

其中, 局部帕累托最优解指不被任何邻域解支配的解; 全局帕累托最优解指不被可行域内任何解支配的解.

为了直观地展示多模态多目标优化问题的特点, 图 2 给出了多模态多目标优化问题的示意图. 如图 2 所示, 左侧是决策空间, 右侧是目标空间, 决策空

间两条曲线 PS_1 和 PS_2 代表两个全局帕累托最优解集，它们映射到目标空间同一个全局帕累托前沿 PF 。决策空间的圆点 A_1 和 A_2 代表两个解集中的可行解，它们都映射到目标空间的圆点 A 。虽然 A_1 和 A_2 在决策空间的距离较大，但是它们在目标空间的距离为零。这是多模态多目标优化问题的一个特点也是求解的难点。第 2 章节会详细介绍此类问题的难点，并综述各种求解算法。

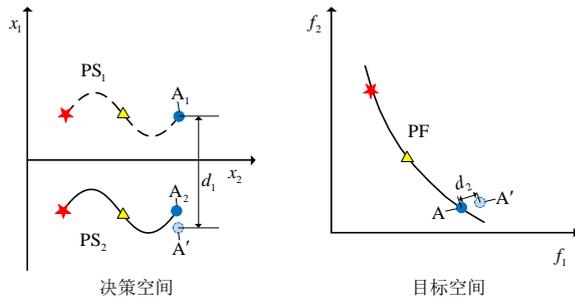


图 2 多模态多目标优化问题示意图

2 多模态多目标优化算法

在智能搜索策略中存在两种关键操作：探索 (Explore) 和开采 (Exploit)。其中探索能力影响收敛速度，开采能力影响搜索到解集的多样性。一般来说，探索能力越强，收敛速度越快，但是容易导致早熟收敛；开采能力越强，搜索到解的多样性越好，但是会减缓收敛速度。优秀的搜索策略需要权衡两种能力的强弱，在提高收敛速度的同时保证解的多样性。在多模态多目标优化中也存在与这两种能力相关的难点。

求解多模态多目标优化问题的难点主要有两个：(1) 在搜索过程中，如何提高算法的搜索能力，使得算法能找到尽可能多的帕累托最优解。(2) 在环境选择中，如何保留目标空间距离较小而决策空间距离较大的解，同时保证决策空间和目标空间的多样性。

难点 (1) 解释：在解决多模态多目标优化问题时，许多算法容易过早收敛到某一个区域。如图 2 所示，算法极易收敛到 PS_1 或 PS_2 中的某一个邻域，比如算法已经搜索到 PS_1 ，那么它就已经找到了一个完整的 PF ，但是决策空间另外一个最优解集 PS_2 还未找到，这极可能造成优良解的丢失。若要提高算法的搜索能力，使得算法能找到尽可能多的帕累托最优解，应当控制信息传递速度，防止算法过快收敛到某个局部区域。

难点 (2) 解释：如图 2 所示，决策空间两个点 A_1 和 A_2 映射到目标空间同一个点 A ，如果算法已经找到点 A_1 和 A' ，且 A' 正在逼近 A_2 ， A' 点很可能在移动到 A_2 前被删除，造成算法难以同时保留 A_1 和 A' ，

因为 A_1 和 A' 在目标空间的距离 d_2 非常小。但是从图中可以看出虽然 A_1 和在目标空间的距离 d_2 非常小，但是它们在决策空间的距离 d_1 相对较大，因此若要同时保留 A_1 和 A' ，不能单纯以目标空间拥挤度距离来衡量它们的优劣，还应考虑决策空间的分布情况。

为了解决以上两大难点，研究者提出了不同的解决方法，因为多模态多目标优化问题解的数量较多，且问题的特性较复杂，演化算法具有求解此类问题的独特优势，因此本文只对多模态多目标演化算法进行介绍，这些方法可以分为提高决策空间多样性的方法和提高搜索能力的方法。

2.1 提高搜索能力的方法

要解决多模态多目标优化问题需要提高算法的搜索能力，使算法能求得尽可能多的帕累托最优解。下面介绍通过提高搜索能力来解决多模态多目标优化问题的算法。

$DN-NSGAI$ ^[13] 在选取父代时，采用决策空间小生境的方法，使种群在不同小生境内搜索不同的帕累托最优解，提高算法的搜索能力。该算法的优点是：促使种群在决策空间多个区域并行进化，防止过早收敛到某一个区域。缺点是：只使用了决策空间拥挤度距离，没有考虑目标空间的分布情况，造成算法得到的帕累托前沿多样性较差，另一个缺点是引入了小生境参数。

为了避免引入小生境参数， $MO_Ring_PSO_SCD$ ^[14] 以环形拓扑结构约束种群信息的传递速度，使种群在搜索过程中逐渐形成稳定的小生境，防止算法过早收敛到某一个区域，使其能够找到更多的帕累托最优解。此外，该算法使用特殊拥挤距离 (SCD) 兼顾决策和目标空间的多样性，克服了 $DN-NSGAI$ 的缺点。该算法的优点是：在不引入任何小生境参数的前提下，促使决策空间形成稳定的小生境，增强了算法的搜索能力，改善了环境选择机制，同时提高了决策空间和目标空间的多样性。此外，在选取父代时，采用二元锦标赛方法，只有决策空间距离相近的粒子才进行相互竞争，这种机制可以防止算法过早收敛到某一个区域。缺点是：在某些 PS 形状复杂的问题上性能不好，种群的分布不够均匀。

$MO_Ring_CSO_SCD$ ^[23] 也采用环形拓扑结构和特殊拥挤度距离来提高算法的搜索能力并改善解的多样性，由于 Competitive Swarm Optimization (CSO) 适用于大规模优化问题，因此该算法在低维问题上优越性不明显。

为了进一步提高解集在决策空间的多样性, MMODE^[24] 采用预选解机制, 选择非支配排序和拥挤度排序较好的解作为父本产生子代; 在产生子代过程中, 如果子代超出边界, 不是直接将它们舍去或将它们直接放在边界上, 而是采用特殊的边界处理机制, 使得出界的子代也有机会继续进化, 这种机制有效的提高了算法的搜索能力. 该算法的优点是: 充分利用已有资源搜索决策空间, 防止出现粒子在边界聚集的现象. 缺点是: 求得的解集在目标空间的分布性欠佳.

MO_PSO_MM^[25] 和 MMOPIO^[26] 都采用自组织映射方法将原始决策空间转化到映射空间, 根据粒子在映射空间的分布确定它们的邻域关系, 使种群在映射空间邻域内进化, 提高算法的搜索能力. 这两种算法的优点是: 可以将高维决策空间映射到一维或二维空间, 便于可视化粒子的邻域关系. 缺点是: 映射机理不清晰, 随机性较大.

使用存档机制, 充分挖掘种群在决策空间的信息, 也是求解多模态多目标优化的一种有效方法. TriMOEA-TA&R^[22] 首先通过决策变量分析技术检测出与收敛相关的决策变量, 并使用收敛性存档优化这些变量, 提高目标空间的收敛性. 为了提高多样性, 作者采用了多样性存档, 多样性档案旨在维持目标空间和剩余决策子空间的多样性 (即从总搜索空间中移除收敛相关决策子空间). 最后, 通过重新组合选出两种存档中的优秀解, 构建具有良好整体性能的最终解集. 该算法的优点是: 使用两个外部存档, 分别负责改善多样性和收敛性, 且兼顾决策空间和目标空间的多样性. 缺点是: 当不同帕累托最优解之间的距离相差较大时, 算法性能下降, 且在产生参考向量时引入了新的参数. DNEA-L^[27] 是通过改进 DNEA^[28] 得到的, DNEA-L 使用多前沿存档来存储多个帕累托前沿, 以提高算法同时保留全局最优和局部最优解的能力; 环境选择中采用 double-sharing 的适应度分享函数值作为筛选指标, 同时改善决策空间和目标空间的多样性. 该算法的优点是: 能够同时保留全局和局部最优解集, 能够得到决策空间和目标空间分布性较好的解. 缺点是: 引入了多个参数, 比如适应度值分享半径, 需要保留非支配前沿的个数等.

MOEA/D-AD^[29] 把多模态多目标优化问题分解成多个子问题, 每个子问题由一个或者几个个体进行寻优, 提高了解集在决策空间的多样性. 该方法的优点是: 可以保留目标空间距离小但决策空间距离

大的解. 缺点是: 引入了确定邻域大小的参数.

强化学习或机器学习与进化算法结合是求解多模态多目标优化问题的另一途径. DE-RLFR^[30] 将强化学习机制融入差分进化算法来解决多模态多目标优化问题. 首先根据个体的适应度值划分不同的等级, 不同的等级有不同的进化方法, 根据进化后子代与父代等级高低设置奖励级别. 该方法的优点是: 进化过程向邻近最优等级逐步进行, 保证收敛的稳定性. 缺点是引入了等级划分参数, 且在高目标优化中奖励情况较复杂. SS-MOPSO^[31] 采用自组织 Speciation 策略将种群分成不同的子种群, 不同种群之间不存在重叠, 促使算法找到更多帕累托解. MMO-Clustering-PSO^[32] 通过聚类把整个种群分成多个子种群, 子种群内的粒子根据自己的全局最优更新位置和速度, 不同种群之间使用环形拓扑结构进行信息传递. 此方法的优点是搜索能力较强, 能找到较多的帕累托最优解. 缺点在于不同问题中聚类数目难以设定, 且该参数对算法性能影响较大.

以上方法通过提高搜索能力防止算法陷入局部区域, 对解决多模态多目标优化问题有很大的帮助. 然而只有搜索机制是不够的, 还需要有较好的环境选择机制来提高决策空间的多样性, 保留决策空间距离大但目标空间距离小的解, 进而更好地解决多模态多目标优化问题.

2.2 提高决策空间多样性的方法

上述提高搜索能力的方法可以使算法找到更多的帕累托解, 除此之外, 多模态多目标优化问题需要特殊的环境选择方法才能使目标空间距离相近而决策空间距离较远的帕累托最优解 (如图 2 中 A_1 和 A_2) 都被保留下来. 以下是提高决策空间多样性的相关工作.

Deb et al.^[15] 提出 Omni-optimizer 算法, 在该算法的环境选择机制中, 根据非支配排序、决策空间拥挤度距离和目标空间拥挤度距离三个指标对个体进行排序. 其中非支配排序方法如图 3 所示, 帕累托最前沿的点集为第 1 前沿, 帕累托次前沿的点集为第 2 前沿, 以此类推. 决策空间拥挤度距离的计算如图 4 所示. 以二维决策空间问题为例, 计算拥挤度距离在 x_1 方向的分量时, 首先根据 x_1 的值由小到大排序, 粒子的拥挤度距离等于该粒子前后邻居相应决策变量的差值; 边界粒子 (图中第 5 个点) 的拥挤度距离等于它与最近邻居相应决策变量差值的二倍. 目标空间拥挤度距离计算方式与决策空间类似, 只有边界点的处理不同, 决策空间边界点拥挤度距离为与最

近邻居差值的二倍, 而目标空间边界点拥挤度距离取无穷大. 最后根据条件选择两个空间的拥挤距离之一分配给对应粒子. 条件选择后的拥挤度距离能反映粒子决策空间和目标空间邻域的拥挤度, 进而同时保证决策空间和目标空间的多样性. 之后许多方法(如 MO_Ring_PSO_SCD^[14]、MO_PSO_MM^[25]等)都采用了兼顾决策和目标空间拥挤度距离的环境选择机制.

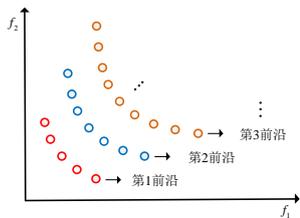


图3 非支配排序

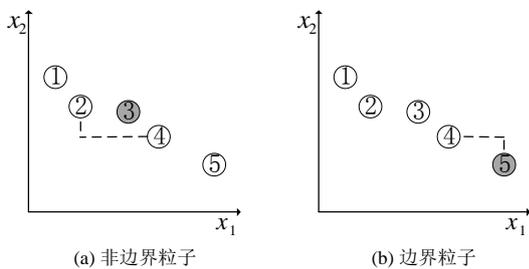


图4 决策空间拥挤度距离的计算方法

Liang et al.^[13] 提出一种决策空间小生境策略, 能在环境选择中帮助保留更多帕累托最优解. 该策略首先根据非支配关系将种群分成不同的帕累托前沿, 然后依据 Omni-optimizer 中决策空间拥挤度距离把同一个前沿内的粒子进行排序. Yan et al.^[33] 提出 PEN-MOBA 算法, 在决策空间引入小生境策略, 并增加了局部最优外部存档来提高决策空间多样性. 但是这两种选择机制没有考虑目标空间的分布情况.

为了兼顾决策空间和目标空间的分布情况, Yue et al.^[14] 提出一种特殊拥挤度距离. 在环境选择中, 首先根据非支配关系排列出不同的前沿, 在同一前沿中的粒子再根据特殊拥挤度距离由大到小排序, 特殊拥挤度距离的计算公式如下:

$$SCD_i = \begin{cases} \text{Max}(CD_{i,x}, CD_{i,f}) & \text{if } CD_{i,x} > CD_{avg,x} \\ & \text{or } CD_{i,f} > CD_{avg,f} \\ \text{Min}(CD_{i,x}, CD_{i,f}) & \text{otherwise} \end{cases} \quad (2)$$

其中 SCD_i 表示第 i 个粒子的特殊拥挤度距离, $CD_{i,x}$ 表示第 i 个粒子在决策空间的拥挤度距离, $CD_{i,f}$ 表示第 i 个粒子在目标空间的拥挤度距离, $CD_{avg,x}$ 表示决策空间的平均拥挤度距离, $CD_{avg,f}$ 表示目标空间的平均拥挤度距离. 由公式 (2) 可看出, 当第 i 个

粒子的决策空间或目标空间拥挤度距离大于平均水平时, SCD 取较大的一个, 否则取较小的一个. 图 2 中 A_1 和 A' 的特殊拥挤度距离取决策空间的值, 因此在环境选择中二者被同时保留的可能性增大. 环境选择中使用非支配关系和特殊拥挤度距离排序可以提高决策空间和目标空间的多样性. 文献 [24-26] 也采用了这种环境选择机制, 均取得了较好的效果.

将多样性指标嵌入目标函数是改善种群多样性的一种有效方法. Liu et al.^[27] 设计了 double-sharing 函数, 该函数将决策空间和目标空间适应度值分享后组合到一个函数中, 并将组合后的函数值用于环境选择标准, 以达到同时改善决策空间和目标空间多样性的目的.

大部分小生境方法都基于欧式距离确定邻域关系, 但在较高维空间欧式距离确定的邻域会出现偏差, Shi et al.^[34] 提出 MMOEA-GD 算法, 该算法在决策空间使用谐波平均距离 (Harmonic Average Distance) 衡量种群在决策空间的邻域关系, 只允许粒子与决策空间邻域内的个体进行信息交互, 从而提高种群在决策空间的多样性.

也有学者使用聚类方法辅助改善种群的多样性. Liu et al.^[22] 提出一种基于小生境的清除策略, 该策略首先根据一组分布均匀的参考向量把种群进行聚类, 以改善目标空间的均匀性和延展性; 对每个子类采用小生境方法促进决策空间的多样性. Maity et al.^[35] 提出 MM-NAEMO 算法, 它由 NAEMO 改进而来, 改进后的环境选择中使用了基于混合高斯模型的聚类算法, 先把种群聚成不同的子类, 然后在每个子类中挑选优秀个体, 以改善种群的多样性.

上述算法的信息总结见表 1. 表中给出了各种算法的基本框架和有效机制. 这些算法的代码可以在多模态多目标优化问题相关研究主页下载 (<http://www5.zzu.edu.cn/cilab/MMO.htm>).

2.3 典型算法举例

以上两个小节从提高算法搜索能力和改善决策空间多样性两个方面对相关工作进行了介绍, 为了说明求解此类问题的整体流程, 本节给出典型算法的求解过程.

使用环形拓扑结构和特殊拥挤距离的多目标粒子群优化算法 (MO_Ring_PSO_SCD)^[14] 是求解多模态多目标优化问题的典型算法, 且算法同时具有提高搜索能力以及改善决策空间多样性机制, 因此选择它来说明求解多模态多目标优化问题的整体流程.

表 1 多模态多目标优化算法概况

算法名称	基本框架	有效机制
1 Omni-optimizer ^[15]	遗传算法	环境选择中同时使用决策空间和目标空间拥挤度距离.
2 MMOEA-GD ^[34]	差分进化算法	环境选择中使用决策空间谐波平均距离.
3 MM-NAEMO ^[35]	遗传算法	环境选择中使用基于混合高斯模型的聚类方法.
4 PEN-MOBA ^[33]	蝙蝠算法	搜索过程中使用决策空间小生境策略; 使用局部最优存档.
5 DN-NSGAI ^[13]	遗传算法	搜索过程中使用决策空间小生境策略; 环境选择中使用决策空间拥挤度距离.
6 DNEA-L ^[27]	遗传算法	搜索过程中使用多前存档; 环境选择中使用 double-sharing 同时提高决策空间和目标空间多样性.
7 TriMOEA-TA&R ^[22]	基于分解的进化算法	搜索过程中使用收敛和多样性两个存档; 环境选择中使用聚类和大小生境策略分别提高目标空间和决策空间多样性.
8 MOEA/D-AD ^[29]	基于分解的进化算法	把分解后的单目标优化子问题当做多模态单目标问题解决.
9 DE-RLFR ^[30]	差分进化算法	采用强化学习机制引导算法逐步收敛.
		搜索过程中有效机制
10 MO_Ring_PSO_SCD ^[14]	粒子群优化算法	使用特殊拥挤度距离提高目标空间和决策空间多样性.
11 MO_Ring_CSO_SCD ^[23]	竞争粒子群优化算法	
12 MMODE ^[24]	差分进化算法	
13 MO_PSO_MM ^[25]	粒子群优化算法	
14 MMOPIO ^[26]	鸽群优化算法	
15 MO-Clustering-PSO	粒子群优化算法	
16 SS-MOPSO ^[31]	粒子群优化算法	使用自组织 Speciation 构建多个子种群.

MO_Ring_PSO_SCD 的整体流程如算法 1 所示. 首先初始化种群 P 、个体最优存档 PBA 和邻域最优存档 NBA (1-6 行); 然后根据非支配关系和特殊拥挤距离对 PBA 和 NBA 中的个体进行优劣排序 (11-12 行), 按照非支配排序由前到后, 特殊拥挤距离由大到小进行排序; 选择排序后 PBA 和 NBA 中的第一个个体作为当前个体的个体最优和邻域最优, 并根据粒子群优化算法个体位置和速度的更新公式对当前粒子进行更新 (14-19 行); 把更新后的粒子放入个体最优存档, 删除存档中被更新后个体所支配的所有个体 (21 行); 使用当前个体最优存档、前一个体最优存档以及后一个体最优存档组合成暂时邻域最优存档 ($temp_NBA\{i\}$), 从暂时邻域最优存档中选出所有非支配个体作为更新后的邻域最优存档 (24-33 行); 循环上述更新过程直至满足停止迭代条件; 最后输出 NBA 中的非支配解集.

经过分析发现, MO_Ring_PSO_SCD 算法针对多模态多目标优化问题的有效处理机制有两个: (1) 使用环形拓扑结构; (2) 使用特殊拥挤距离. 环形拓扑结构将整个种群分成多个小生境, 不同小生境之间不存在直接竞争, 大大提高算法找到多个最优解的能力; 特殊拥挤距离兼顾决策空间和目标空间拥挤距离, 使得算法能够保留目标空间拥挤距离小但决策空间拥挤距离大的解. 关于两种机制的具体描述及有效性验证请参考 [14].

通过对各种多模态多目标进化算法进行分析总结发现此类算法的一般框架如图 5 所示, 它和一般进化优化算法的总体流程一致, 首先, 初始化一个种群; 种群中不同的个体根据环境变化或其他个体分享的信息不断调整进化方向; 然后, 根据预定义的算子对种群进行筛选; 如此迭代式地对个体进行

调整筛选, 直至满足预设的终止条件. 多模态多目标优化方法的独特之处在于: 进化过程中需加入限制信息传递范围的策略, 防止种群过早收敛到局部区域; 在筛选过程中需要兼顾决策和目标空间的多样性, 使得多模态解能在环境选择中生存下来.

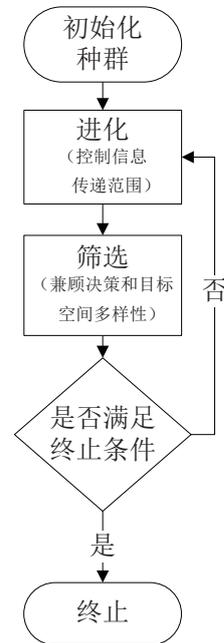


图 5 多模态多目标进化算法一般框架

3 测试函数和评价指标

为了公平、公开、合理地比较不同多模态多目标优化算法, 学者们提出不同的测试函数集和能够衡量算法性能优劣的评价指标来评价算法的性能.

算法 1: MO_Ring_PSO_SCD

```

1 //初始化种群 P
2 Evaluation(P)
3 //初始化个体最优和邻域最优存档
4
5 PBA = P
6 NBA = PBA
7
8 while Generation < MaxGenerations do
9   for i = 1 : ParticleNumber
10    //对 PBA 和 NBA 中的个体进行排序
11    Sorted_PBA{i} = non-dominated_scd_sort(PBA{i})
12    Sorted_NBA{i} = non-dominated_scd_sort(NBA{i})
13    //选择 pbest 和 nbest
14    pbesti ← Sorted_PBA{i} 中的第一个
15    nbesti ← Sorted_NBA{i} 中的第一个
16    //更新种群 P
17    Pi(t + 1) = Pi(t) + vi(t + 1)
18    vi(t + 1) = Wvi(t) + C1r1(pbesti - Pi(t))
19    + C2r2(nbesti - Pi(t))
20    Evaluation(Pi(t + 1))
21    //更新 PBA
22    把 Pi(t + 1) 放入 PBA{i} 并删除被 Pi(t + 1) 支配的个体
23    end for
24    //更新 NBA
25    for i = 1 : ParticleNumber
26    if i = 1
27      temp_NBA{i} = [PBA { ParticleNumber }, PBA {1}, PBA]
28    else if i = ParticleNumber
29      temp_NBA{i} = [PBA { ParticleNumber - 1},
30      PBA { ParticleNumber }, PBA {1}]
31    else
32      temp_NBA{i} = [PBA {i-1}, PBA {i}, PBA {i+1}]
33    end if
34    NBA {i} = non-dominated particles in temp_NBA{i}
35    end for
36  end while
37 输出: NBA 中的非支配解集

```

3.1 测试函数

标准测试函数集可以用来公平测试不同算法的性能, 因为测试问题的各项信息都已知, 所以可以避免实际问题中不确定因素造成的影响, 这对相关领域的研究具有促进作用. 因此, 多模态多目标优化也需要标准测试函数, 这些函数与普通多目标测试函数不同, 它们具有多个全局或局部 PS.

1999 年 Deb^[37] 最早提到了多目标优化问题中的多模态情形, 并给出了 Omni-test 测试函数, 该函数的决策变量的维度和等效 PS 的个数都可扩展, 但是目标空间的维度不可扩展. Rudolph et al.^[38] 提出了 SYM-PART 测试函数, 且在函数中加入了旋转和扭曲的操作, 使不同决策变量之间存在相关性.

Preuss et al.^[39] 提出了 TWO-ON-ONE 测试函数, 用以分析传统进化多目标优化算法在多模态多目标优化问题上的性能. Ishibuchi et al.^[40] 提出了 Polygon-based problem, 用于在决策空间直观地显示种群的多样性, 但是这些函数决策变量的维度是固定的. 随后, Ishibuchi et al.^[41] 和 Masuda et al.^[42] 提出了决策变量具有可扩展性的多模态多目标测试函数. 此外, Liang et al.^[13] 通过平移和对称操作复制多个 PS 设计了两个多模态多目标测试函数, 这两个测试函数比较简单. 随后, Yue et al.^[14] 设计了六个较为复杂的测试函数, 这些函数具有不同形状的 PS 和 PF, 但是它们决策空间和目标空间的维度不可扩展, 性质较单一. Liang et al. 在文献 [24] 中设计了 PF 和 PS 形状复杂的测试函数, 但是这些函数决策空间和目标空间的维度仍然不可扩展. Liu et al.^[22] 设计了决策空间和目标空间都可扩展的测试函数 MMMOP, 但是这些函数没有局部和全局 PS 共存的情况. Yue et al.^[21] 设计了较为复杂的多模态多目标测试问题 MMF suite, 包括局部 PS 和全局 PS 共存, 形状复杂的 PS 和 PF, 决策变量个数和目标个数可调等特征; 并给出了多模态多目标测试函数的设计方法和设计框架, 这些函数的可扩展性较好.

多模态多目标测试函数期望具有的性质包括: (1) 同时具有局部和全局 PS; (2) PS 的个数可调节; (3) PS 的形状不同; (4) PS 和 PF 已知; (5) 决策变量和目标个数可调. 一般来说, 同时具有局部和全局 PS 的问题比只有全局 PS 的问题难以解决, 因为局部 PS 是陷阱, 搜索能力较弱的算法容易陷入局部 PS. 另外, 决策空间的维度越高, 搜索空间越大, 问题的求解难度越大; 目标的个数越多, 算法的收敛力度减小, 问题求解难度也会增大. 测试函数集既要包括简单的测试函数, 又要包括复杂的测试函数, 这样才能评价不同算法的优劣. 为了方便研究者进行选择, 表 2 对文献中已有多模态多目标优化测试函数的性质进行了总结. 其中 ‘✓’ 表示测试函数集有满足对应性质的函数, ‘✗’ 表示没有满足对应性质的函数, ‘✓/✗’ 表示既有满足对应性质的函数也有不满足对应性质的函数. 如 MMF suite 中既有决策变量个数可调的也有不可调的函数, 因此对应 ‘决策变量个数是否可调’ 的性质为 ‘✓/✗’. 表中所列 Omni-test、SYM-PART 和 TWO-ON-ONE 测试函数是单个测试函数, 因为早期文献中对多模态多目标优化的研究较少, 因此测试函数性质单一; 后面几行所列测试函数均为测试函数集, 每个集合包含多个测试函数, 它们的性质也有所差异. 如果研究者需要测试算法求解超多目标 (3 个以上) 多模态优化问

表 2 多模态多目标测试函数性质

测试函数名称	决策变量个数是否可调	目标个数是否可调	PF 的几何形状	PS 的几何形状	PS 的个数是否可调
Omni-test ^[15]	✓	✗	凸	直线段	✓
SYM-PART ^[38]	✗	✗	凸	直线段	✗
TWO-ON-ONE ^[39]	✗	✗	凸	直线段	✗
Polygon-based problem ^[40]	✓	✓	凸	区域块	✓
Problems in [24]	✗	✗	凸/凹/线性	曲线段	✗
MMMOP ^[22]	✓	✓	线性/凹	直线段/区域块	✗/✓
MMF suite ^[13, 14, 21]	✗/✓	✗/✓	凸/凹, 连续/不连续	曲线(面)段/直线段	✗/✓

题的性能, 建议选择 Polygon-based problem^[40] 或 MMMOP^[22], 因为这两个测试函数集包含目标个数大于 3 的测试函数, 且它们的 PS 几何形状为区域块, 这种测试函数的决策空间只有二维, PS 是多边形内部区域, 通过观察算法求得解对多边形的覆盖情况即可了解算法的性能, 所以这些函数可以实现超多目标优化的可视化. 如果研究者需要测试算法求解具有不同形状 PF 问题的性能, 建议选择 MMF suite, 因为该测试函数集包含多种集合形状的 PF, 且测试函数的其他性质也较丰富.

3.2 评价指标

评价指标可以将不同算法的性能量化, 在算法评估和对比中起着非常重要的作用. 常用的多目标算法的评价指标包括 Inverted Generational Distance (IGD^[43])、HyperVolume (HV^[44]) 等, 它们只能衡量种群在目标空间的分布情况, 对于多模态多目标优化问题而言, 即使目标空间的分布性能良好, 决策空间的性质也可能很差. 因此需要新的评价指标对不同的多模态多目标算法进行评价.

多模态多目标优化算法评价指标需要具有以下性质: (1) 能反映决策空间的多样性、收敛性、覆盖率; (2) 能反映目标空间的多样性、收敛性、覆盖率; (3) 有合理的量化范围; (4) 能用以比较不同算法解决多模态多目标优化问题的能力. 设计具有这些性质的指标具有很多挑战, 所设计的指标需要兼顾多个方面, 只考虑某一个指标不能准确的反应算法的性能. 如图 6 所示, 圆点表示算法得到的解集, 虚线和实线代表问题的两个 PS. 图 6 (a) 中算法得到的解集具有很好的多样性, 但是圆点还没有逼近实线或者虚线, 因此收敛性差; 图 6 (b), 圆点逼近了实线或者虚线, 但是聚成了簇, 因此多样性差; 图 6 (c) 中算法得到的解集具有良好的多样性和收敛性, 但是它们并没有覆盖整个 PS, 而是覆盖了 PS₁ 的一部分, PS₂ 的另外一部分, 虽然这两部分可以组成一个相对

完整的 PF 但事实上算法并没有找到一个完整的 PS. 图 6 (d) 给出了理想情况下, 算法得到解集的分布情况. 由图可得, 同时衡量解集在决策空间和目标空间的分布性、多样性、收敛性和覆盖率是比较困难的.

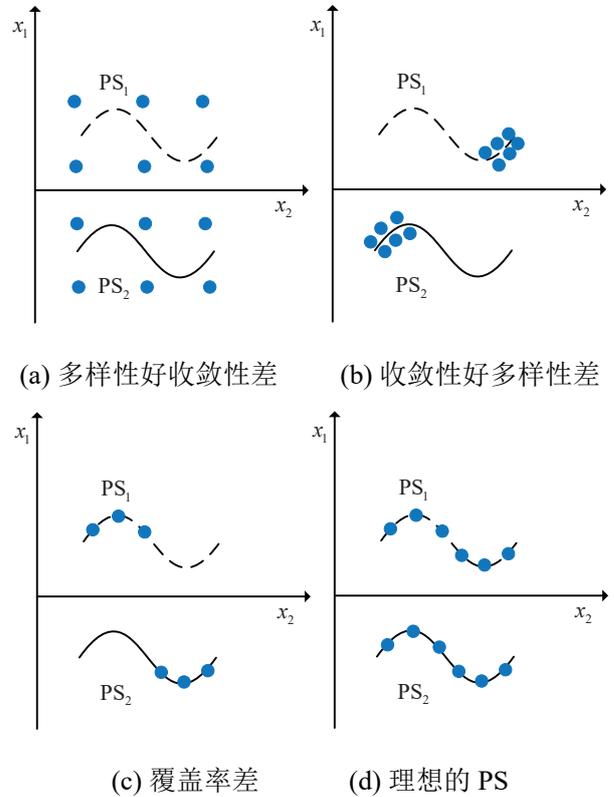


图 6 评价指标设计的难点和挑战

为了解决上述挑战, 研究者提出了不同种类的评价指标. Rudolph et al.^[38] 设计了 Covered Sets (CS) 和 Set Population Spread (SPS) 衡量多模态多目标优化算法的性能. CS 表示帕累托子集被覆盖的数目, CS 越大说明算法求得的解集覆盖帕累托子集的数量越多. SPS 表示接近每个帕累托子集的种群数量的标准差, SPS 越大说明算法求得的解集多样性越好. 但是随机均匀初始化的种群这两个指标值都比较好, 因此这两个指标不能准确的反应算

法的性能. Zhou et al.^[45] 把 IGD ^[43] 用在决策空间 ($IGDX$), 衡量算法求得的 PS 与真实 PS 之间的接近程度. $IGDX$ 的计算方法如下:

令 P^* 代表沿真实 PS 均匀分布的点集 (参考点), O 代表算法求得的解集, $IGDX$ 的计算公式:

$$IGDX(O, P^*) = \frac{\sum_{v \in P^*} d(v, O)}{|P^*|} \quad (3)$$

其中 $d(v, O)$ 表示 v 和 O 中所有点之间欧式距离的最小值, $|P^*|$ 表示参考点的数目. 如果参考点集能够很好的表示真实 PS, 那么 $IGDX$ 就能较好的衡量决策空间的收敛性和多样性. $IGDX$ 的值越小, 说明算法求得的解集和参考点集越接近. 但是只用 $IGDX$ 指标衡量算法性能存在一种缺陷, 当算法求得的解也在真实 PS 上, 但是没有和参考点接近或重合时, $IGDX$ 的值就不能反映求解此问题的情况. 为了更合理的比较不同多模态多目标优化算法的性能, Yue et al.^[14] 把 CR 和 $IGDX$ 结合, 设计了新的衡量指标 PSP . 但是 PSP 的最优值为无穷大, 根据指标值无法判断算法求得解集和最优解集的差距, 于是 Yue et al. 在 [21] 中, 将 PSP 改成 $\frac{1}{PSP}$, 这样评价指标 $\frac{1}{PSP}$ 的最优值变为零, 指标值越接近零说明算法性能越好. $\frac{1}{PSP}$ 的计算方法如公式 (4). 它既能反映算法求得解集对真实 PS 的覆盖率, 又能反映它们之间的接近程度.

$$\frac{1}{PSP} = \frac{IGDX}{CR} \quad (4)$$

其中 (CR) 表示算法求得的解集对真实解集的覆盖率. $\frac{1}{PSP}$ 越小说明算法求得的解集与真实解集越接近, 且算法求得的解集对真实解的覆盖率越大. $\frac{1}{PSP}$ 也只能反应决策空间的性质, 为了综合评价算法的性能, 需要搭配目标空间衡量指标 (如 HV ^[44]) 对算法进行综合评价.

为了同时评价解集在决策空间和目标空间的性质, Liu et al.^[22] 提出新的评价指标 $IGDM$, 该指标可以衡量解集在决策空间和目标空间的收敛性和多样性. 令 $F^* : \{f_1^*, f_2^*, \dots, f_q^*\}$ 为 PF 上均匀分布的参考点, 其中 q 表示参考点的个数; $A : \{a_1, a_2, \dots, a_q\}$, 其中 a_i ($i = 1, 2, \dots, q$) 是目标值为 f_i^* 的帕累托最优解的个数; $X^* : \{x_{1,1}^*, x_{1,2}^*, \dots, x_{1,a_1}^*, x_{2,1}^*, \dots, x_{q,a_q}^*\}$ 代表与 F^* 和 A 对应的帕累托最优解集; P 是算法求得的解集, $IGDM$ 的定义如下:

$$IGDM(P, F^*, X^*) = \frac{\sum_{f_i^* \in F^*} \sum_{j=1, \dots, x_p} d(f_i^*, x_{i,j}^*, P)}{|X^*|} \quad (5)$$

其中 $d(f_i^*, x_{i,j}^*, P) = \begin{cases} d_{\max} & \text{if } P_{i,j} = \emptyset \\ \min \{d_{\max}, ed(f_i^*, P_{i,j})\} & \text{else} \end{cases}$

$P_{i,j} = \{x_k : j = \arg \min_{l=1, \dots, a_i} ed(x_{i,l}^*, x_k), x_k \in P\}$ $ed(f_i^*, P_{i,j})$ 表示 F^* 和 $P_{i,j}$ 之间的最小欧式距离; $ed(x_{i,l}^*, x_k)$ 表示 $x_{i,l}^*$ 和 x_k 之间的最小欧式距离, d_{\max} 是预定义的参数, 一般设置为 1.

虽然已经有一些多模态多目标优化算法的评价指标, 但是大部分都需要已知问题的 PS 和 PF, 并根据它们选取参考点或参考向量. 而在实际问题中 PS 和 PF 是未知的, 算法性能难以评价. 因此, 多模态多目标优化中, 在未知 PS 和 PF 的情况下, 如何设计合理的性能指标是未来研究的一个方向. 下一部分介绍多模态多目标优化的其他未来研究方向和应用领域.

4 应用领域及未来研究方向

4.1 应用领域

多模态多目标优化具有广泛的应用领域, 如桥梁优化问题^[46], 饮食设计问题^[38], 空间任务设计问题^[47], 化工生产优化问题^[48], 火箭发动机任务设计问题^[49], 功能性脑成像问题^[50], 路径规划问题^[14], 特征选择问题^[17], 建筑选址问题^[40] 等. 上述领域的多目标优化问题具有典型的多模态特性, 且不同的解集具有不同的实际意义. 例如某个路径规划问题以路程最短和红绿灯个数最少为优化目标, 存在路程和红绿灯个数都相同的多条路径, 但是它们途径的公共设施可能大有区别, 如果使用多模态多目标优化算法找到多个符合要求的路径, 那么不同的司机可以根据自己对公共设施的偏好选择不同的路径. 由于多模态多目标优化的发展仍处于初级阶段, 虽然许多问题属于多模态多目标优化问题, 但是技术人员并未采用多模态多目标优化算法解决这些问题. 我们需要开发出高效稳定的算法, 为决策者提供更多选择, 也为节约能源, 减少污染, 降低成本做出贡献.

4.2 未来研究方向

多模态多目标优化领域未来研究方向包括以下几个:

第一, 如何同时找到全局和局部帕累托最优解集是多模态多目标优化中的未来研究方向之一. 通过本文的分析可知, 根据全局和局部最优解集的情况, 多模态多目标优化问题可分为两类: 第一类问题只具有全局帕累托最优解集, 第二类问题既有全局帕累托最优解集又有局部帕累托最优解集. 现有的多模态多目标优化算法大多数用来解决第一类问题, 对于第二类问题的解决效果不佳. 因此, 如何搜索到第二类问题的全局和局部最优解集是未来值得研究

的方向。

第二, 决策空间收敛性和多样性不平衡问题^[51]是多模态多目标优化领域新的挑战。在决策空间收敛性和多样性不平衡问题中, 有些帕累托最优解集极易搜索到, 但有些解集则极难找到, 因为不同解集附近的适应度地形不同, 如果某个解集所在的盆区较大, 则搜索此解集的难度较小, 如果某个解集所在的盆区较狭窄或较崎岖, 那么搜索此解集的难度会较大。如何处理多模态多目标优化中收敛性和多样性不平衡问题也是未来的研究方向之一。

第三, 多模态多目标优化问题最终解的选择也是待解决的难题。一个帕累托最优解集包含多个可行解, 多个帕累托解集包含的可行解数目大大增加, 而在一个实际应用的具体情况下, 必须从这些解中挑选出一个使用, 不同决策者在不同情况下会挑选不同的解, 因此最终解的选择是未来研究的难题。

第四, 离散多模态多目标优化问题也是未来研究的方向之一^[52]。特征选择、路径规划、车间流水线调度中存在大量的多模态多目标优化问题, 但它们都属于离散组合优化问题, 本文介绍的算法均是连续优化算法, 不能直接用来求解离散优化问题。如何解决离散多模态多目标优化中编码难题及进化导向问题是未来需要研究的方向。

除了上述方向以外, 在该领域还有许多待研究的问题, 包括: 动态、高维、高目标多模态优化问题研究^[53, 54], 多模态多目标优化中高维可视化技术研究, 多模态多目标优化收敛性分析等理论研究。

5 总结与展望

本文对多模态多目标优化领域的相关工作进行了分析总结, 从提高算法搜索能力和提高决策空间多样性两个方面介绍了相关算法, 阐述了解决多模态多目标优化问题的困难和挑战。提高算法搜索能力即增强探索能力, 提高决策空间多样性即增强开采能力, 这两方面能力的权衡是解决多模态多目标优化问题的关键。

多模态多目标优化是进化计算领域相对新的研究方向, 它具有广阔的应用前景, 该方向还有许多未解决的问题值得研究者开展深入研究。

参考文献 (References)

[1] Qu B, Suganthan P N. Novel multimodal problems and differential evolution with ensemble of restricted tournament selection[C]. proceedings of the IEEE Congress on Evolutionary Computation, 2010: 1-7.
 [2] Qu B, Suganthan P N, Das S. A distance-based locally informed particle swarm model for multimodal

optimization[J]. IEEE Transactions on Evolutionary Computation, 2013, 17(3): 387-402.
 [3] Li X, Epitropakis M G, Deb K, Engelbrecht A. Seeking Multiple Solutions: An Updated Survey on Niching Methods and Their Applications[J]. IEEE Transactions on Evolutionary Computation, 2017, 21(4): 518-538.
 [4] Li X. Niching without niching parameters: Particle swarm optimization using a ring topology[J]. IEEE Transactions on Evolutionary Computation, 2010, 14(1): 150-169.
 [5] 李宝磊, 施心陵, 苟常兴, 吕丹桔, 安镇宙, 张榆锋. 多元优化算法及其收敛性分析 [J]. 自动化学报, 2015, 41(5): 949-959.
 (Li B, Shi X, Gou C, LÜ D, AN Z, Zhang Y Multivariate optimization algorithm and its convergence analysis[J]. ACTA Automatica Sinica, 2015, 41(5): 949-959.)
 [6] 张贵军, 陈铭, 周晓根. 动态小生境半径两阶段多模态差分进化算法 [J]. 控制与决策, 2016, 31(7): 1185-1191.
 (Zhang G, Chen M, Zhou X Two-stage differential evolution algorithm using dynamic niche radius for multimodal optimization[J]. Control and Decision, 2016, 31(7): 1185-1191.)
 [7] Konak A, Coit D W, Smith A E. Multi-objective optimization using genetic algorithms: A tutorial [J]. Reliability Engineering & System Safety, 2006, 91(9): 992-1007.
 [8] 肖俊明, 周谦, 瞿博阳, 韦学辉. 多目标进化算法及其在电力环境经济调度中的应用综述 [J]. 郑州大学学报 (工学版), 2016, 2016(2): 1-9.
 (Xiao J, Zhou Q, Qu B, Wei X Multi-objective evolutionary algorithm and its application in electric power environment economic dispatch[J]. Journal of Zhengzhou University(Engineering Science), 2016, 2016(2): 1-9.)
 [9] 张勇, 巩敦卫, 郝国生, 蒋余庆. 含区间参数多目标系统的微粒群优化算法 [J]. 自动化学报, 2008, 34(8): 921-928.
 (Zhang Y, Gong D, Hao G, Jiang Y Particle swarm optimization for multi-objective systems with interval parameters[J]. ACTA Automatica Sinica, 2008, 34(8): 921-928.)
 [10] 丁进良, 陈佳鑫, 马欣然. 基于自适应差分进化的常压塔轻质油产量多目标优化 [J]. 控制与决策, 2020, 35(3): 95-103.
 (Ding J, Chen J, Ma X Multi-objective optimization of light oil production in atmospheric distillation column based on self-adaptive differential evolution[J]. Control and Decision, 2020, 35(3): 95-103.)
 [11] 陈黄科, 伍国华, 霍离俗, 戚玉涛. 基于目标空间划分的自适应多目标进化算法 [J]. 软件学报, 2018, 2018(9): 2649-2663.
 (Chen H, Wu G, Huo L, Qi Y Objective space division based adaptive multiobjective optimization algorithm[J]. Journal of Software, 2018, 2018(9): 2649-2663.)

- [12] Chen H, Wu G, Pedrycz W, Suganthan P N, Xing L, Zhu X. An adaptive resource allocation strategy for objective space partition-based multiobjective optimization[J]. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2019, 1-16.
- [13] Liang J J, Yue C, Qu B. Multimodal multi-objective optimization: A preliminary study[C]. *proceedings of the IEEE Congress on Evolutionary Computation, Vancouver, 2016: 2454-2461.*
- [14] Yue C, Qu B, Liang J J. A multiobjective particle swarm optimizer using ring topology for solving multimodal multiobjective problems[J]. *IEEE Transactions on Evolutionary Computation*, 2018, 22(5): 805-817.
- [15] Deb K, Tiwari S. Omni-optimizer: A procedure for single and multi-objective optimization[C]. *proceedings of the International Conference on Evolutionary Multi-Criterion Optimization, 2005: 47-61.*
- [16] 许伟伟, 梁静, 岳彩通, 瞿博阳. 多模态多目标差分进化算法求解非线性方程组 [J]. *计算机应用研究*, 2019, 36(5): 1-7.
(Xu W, Liang J, Yue C, Qu B Multimodal multi-objective differential evolution algorithm for solving nonlinear equations[J]. *Application Research of Computers*, 2019, 36(5): 1-7.)
- [17] Yue C, Liang J J, Qu B, Yu K, Song H. Multimodal multiobjective optimization in feature selection[C]. *proceedings of the IEEE Congress on Evolutionary Computation, 2019: 302-309.*
- [18] Xue B, Zhang M, Browne W N. Particle swarm optimization for feature selection in classification: a multi-objective approach[J]. *IEEE Transactions on Cybernetics*, 2013, 43(6): 1656-1671.
- [19] Kamyab S, Eftekhari M. Feature selection using multimodal optimization techniques[J]. *Neurocomputing*, 2016, 171(2016): 586-597.
- [20] Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multiobjective genetic algorithm: NSGA-II[J]. *IEEE Transactions Evolutionary Computation*, 2002, 6(2): 182-197.
- [21] Yue C, Qu B, Yu K, Liang J J, Li X. A novel scalable test problem suite for multimodal multiobjective optimization[J]. *Swarm and Evolutionary Computation*, 2019, 48(2019): 62-71.
- [22] Liu Y, Yen G G, Gong D. A multi-modal multi-objective evolutionary algorithm using two-archive and recombination strategies[J]. *IEEE Transactions on Evolutionary Computation*, 2018, 23(4): 660-674.
- [23] Wang Y, Yang Z L, Guo Y J, Zhu J C, Zhu X D. A novel multi-objective competitive swarm optimization algorithm for multi-modal multiobjective problems[C]. *proceedings of the IEEE Congress on Evolutionary Computation, 2019: 271-278.*
- [24] Liang J J, Xu W W, Yue C T, Yu K J, Song H, Crisalle O D, Qu B Y. Multimodal multiobjective optimization with differential evolution[J]. *Swarm and Evolutionary Computation*, 2018, 44(2019): 1028-1059.
- [25] Liang J J, Guo Q, Yue C, Qu B, Yu K. A self-organizing multi-objective particle swarm optimization algorithm for multimodal multi-objective problems[C]. *proceedings of the International Conference on Swarm Intelligence, 2018: 550-560.*
- [26] Hu Y, Wang J, Liang J J, Yu K, Song H, Guo Q, Yue C, Wang Y. A self-organizing multimodal multi-objective pigeon-inspired optimization algorithm[J]. *SCIENCE CHINA Information Sciences*, 2019, 62(70206): 1-17.
- [27] Liu Y, Ishibuchi H, Nojima Y, Masuyama N, Han Y. Searching for local pareto optimal solutions: a case study on polygon-based problems[C]. *proceedings of the IEEE Congress on Evolutionary Computation, 2019: 896-903.*
- [28] Liu Y, Ishibuchi H, Nojima Y. A double-niched evolutionary algorithm and its behavior on polygon-based problems[C]. *proceedings of the International Conference on Parallel Problem Solving from Nature, 2018: 262-273.*
- [29] Tanabe R, Ishibuchi H. A decomposition-based evolutionary algorithm for multi-modal multi-objective optimization[C]. *proceedings of the International Conference on Parallel Problem Solving from Nature, Coimbra, 2018: 249-261.*
- [30] Li Z, Shi L, Yue C, Shang Z, Qu B. Differential evolution based on reinforcement learning with fitness ranking for solving multimodal multiobjective problems[J]. *Swarm and Evolutionary Computation*, 2019, 49(2019): 234-244.
- [31] Qu B, Li C, Liang J, Yan L, Yu K, Zhu Y. A self-organized speciation based multi-objective particle swarm optimizer for multimodal multi-objective problems[J]. *Appl Soft Comput*, 2019, 105886: 1-13.
- [32] Zhang W, Li G, Zhang W, Liang J, Yen G G. A cluster based PSO with leader updating mechanism and ring-topology for multimodal multi-objective optimization[J]. *Swarm and Evolutionary Computation*, 2019, 50(2019): 100569: 1-13.
- [33] Yan L, Li G S, Jiao Y C, Qu B Y, Yue C T, Qu S K. A performance enhanced niching multi-objective bat algorithm for multimodal multi-objective problems[C]. *proceedings of the IEEE Congress on Evolutionary Computation, 2019: 1275-1282.*
- [34] Shi R Z, Lin W, Lin Q Z, Zhu Z X, Chen J Y. Multimodal multi-objective optimization using a density-based one-by-one update strategy[C]. *proceedings of the IEEE Congress on Evolutionary Computation, 2019: 295-301.*
- [35] Maity K, Sengupta R, Sha S. MM-NAEMO: Multimodal neighborhood-sensitive archived evolutionary many-objective optimization algorithm[C]. *proceedings of the IEEE Congress on Evolutionary Computation, 2019: 286-294.*
- [36] Villalobos C a R, Coello C a C. A new multi-objective evolutionary algorithm based on a performance assessment indicator[C]. *proceedings of the Conference*

- on Genetic and Evolutionary Computation, 2012: 505-512.
- [37] Deb K. Multi-objective genetic algorithms: Problem difficulties and construction of test problems[J]. Evolutionary Computation, 1999, 7(3): 205-230.
- [38] Rudolph G, Naujoks B, Preuss M. Capabilities of EMOA to detect and preserve equivalent pareto subsets[C]. proceedings of the International Conference on Evolutionary Multi-Criterion Optimization, Berlin, 2007: 36-50.
- [39] Preuss M, Naujoks B, Rudolph G. Pareto Set and EMOA behavior for simple multimodal multiobjective functions[C]. proceedings of the Parallel Problem Solving from Nature, Berlin, 2009: 513-522.
- [40] Ishibuchi H, Akedo N, Nojima Y. A many-objective test problem for visually examining diversity maintenance behavior in a decision space[C]. proceedings of the Conference on Genetic and Evolutionary Computation, Dublin, 2011: 649-656.
- [41] Ishibuchi H, Yamane M, Akedo N, Nojima Y. Many-objective and many-variable test problems for visual examination of multiobjective search[C]. proceedings of the IEEE Congress on Evolutionary Computation, Cancun, 2013: 1491-1498.
- [42] Masuda H, Nojima Y, Ishibuchi H. Visual examination of the behavior of EMO algorithms for many-objective optimization with many decision variables[C]. proceedings of the IEEE Congress on Evolutionary Computation, Beijing, 2014: 2633-2640.
- [43] Zhang Q, Zhou A, Jin Y. RM-MEDA: A regularity model-based multiobjective estimation of distribution algorithm[J]. IEEE Transactions on Evolutionary Computation, 2008, 12(1): 41-63.
- [44] Zitzler E, Thiele L, Laumanns M, Fonseca C M, Da Fonseca V G. Performance assessment of multiobjective optimizers: An analysis and review[J]. IEEE Transactions on Evolutionary Computation, 2003, 7(2): 117-132.
- [45] Zhou A, Zhang Q, Jin Y. Approximating the set of pareto-optimal solutions in both the decision and objective spaces by an estimation of distribution algorithm[J]. IEEE Transactions on Evolutionary Computation, 2009, 13(5): 1167-1189.
- [46] Ulrich T. Pareto-Setanalysis: biobjective clustering in decision and objective spaces[J]. Journal of Multi-Criteria Decision Analysis, 2013, 20(5-6): 217-234.
- [47] Schütze O, Vasile M, Coello Coello C A. Computing the set of epsilon-efficient solutions in multiobjective space mission design[J]. Journal of Aerospace Computing, Information, and Communication, 2011, 8(3): 53-70.
- [48] 钱锋, 杜文莉, 钟伟民, 唐漾. 石油和化工行业智能优化制造若干问题及挑战[J]. 自动化学报, 2017, 43(6): 893-901.
(Qian Feng, Du W, Zhong W, Tang Y Problems and challenges of smart optimization manufacturing in petrochemical industries[J]. ACTA Automatica Sinica, 2017, 43(6): 893-901.)
- [49] Kudo F, Yoshikawa T, Furuhashi T. A study on analysis of design variables in Pareto solutions for conceptual design optimization problem of hybrid rocket engine[C]. proceedings of the IEEE Congress on Evolutionary Computation, New Orleans, 2011: 2558-2562.
- [50] Sebag M, Tarrisson N, Teytaud O, Lefevre J, Baillet S. A Multi-Objective Multi-Modal Optimization Approach for Mining Stable Spatio-Temporal Patterns[C]. proceedings of the International Joint Conference on Artificial Intelligence, Edinburgh, 2005: 859-864.
- [51] Liu Y, Ishibuchi H, Yen G G, Nojima Y, Masuyama N. Handling imbalance between convergence and diversity in the decision space in evolutionary multi-modal multi-objective optimization[J]. IEEE Transactions on Evolutionary Computation, 2019, 24(3): 551-565.
- [52] Liang J J, Yue C, Li G, Qu B, Suganthan P N, Yu K. Problem definitions and evaluation criteria for the cec 2021 on multimodal multiobjective path planning optimization[R], Zhengzhou University and Nanyang Technological University, 2020.
- [53] Qiu W, Zhu J, Wu G, Chen H, Pedrycz W, Suganthan P N. Ensemble many-objective optimization algorithm based on voting mechanism[J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2020, 1-15.
- [54] Chen H, Tian Y, Pedrycz W, Wu G, Wang R, Wang L. Hyperplane assisted evolutionary algorithm for many-objective optimization problems[J]. IEEE Transactions on Cybernetics, 2019, 50(7): 3367-3380.

作者简介

岳彩通(1990—), 男, 讲师, 博士, 从事优化算法、机器学习等研究, E-mail: zzuyuecaitong@163.com;

梁静(1981—), 女, 教授, 博士生导师, 从事智能计算、机器学习及其应用研究, E-mail: liangjing@zzu.edu.cn;

瞿博阳(1984—), 男, 教授, 博士, 从事计算智能及其应用等研究, E-mail: quboyang@zut.edu.cn;

于坤杰(1990—), 男, 副教授, 博士, 从事约束多目标优化及其应用等研究, E-mail: yukunjie@zzu.edu.cn;

王艳丽(1982—), 女, 讲师, 博士, 从事集成学习及其应用等研究, E-mail: wangyanli8210@163.com;

胡毅(1988—), 男, 博士生, 从事优化算法及其应用等研究, E-mail: whoyee@163.com;