

Differential Evolution Using Improved Crowding Distance for Multimodal Multiobjective Optimization

Caitong Yue, P.N. Suganthan, Jing Liang, Boyang Qu, Kunjie Yu, Yongsheng Zhu, Li Yan

摘要

在多目标优化中, 决策空间和目标空间之间的关系通常被假设为一对一的映射, 但并非总是如此。在某些问题中, 不同的变量具有相同或相似的目标值, 这意味着多对一的映射。在这种情况下, 有多个帕累托集合 (Pareto Set, PS) 映射到同一个帕累托前沿 (Pareto Front, PF), 这些问题称为多模态多目标问题 (Multimodal Multiobjective Optimization Problem, MMOP)。该论文提出了一种多模态多目标差分进化算法来解决这些问题。在所提出的方法中, 在考虑决策空间和目标空间的多样性的情况下生成差异向量。在拥挤距离计算过程中, 所有被选中的个体都被考虑在内, 而不是单独考虑每个帕累托等级。决策空间中的拥挤距离被与其邻居的欧氏距离的加权和所取代。在环境选择过程中, 并不是所有帕累托等级靠前的个体都会被选中, 因为其中一些解可能非常拥挤。相反, 一些等级靠后的解很有可能是最优解。通过这些操作, 所提出的算法可以保持多模态多目标优化问题的多个 PS, 并提高决策空间和目标空间的多样性。实验结果表明, 所提方法具有较高的综合性能。

关键词: 多模态多目标优化; 差分进化; 拥挤距离

1 引言

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。在实际应用中存在许多优化问题, 如路径优化^[1], 生产调度优化^[2]等, 早期求解此类问题的方法是求得一个最优解。但是随着科技的发展, 求得的最优解可能会因为实际问题中存在的一些约束条件变得难以实现。因此决策者希望得到多个备选解, 根据实际应用需求选取最符合要求的一个。

多模态优化^[3]可以为决策者提供多个可行方案, 此类优化问题存在多个全局或局部最优解, 因此, 找到多个备选解有助于发现鲁棒性更优的解、揭露问题的潜在性质, 同时, 保留多个最优解可以增加解的多样性。但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。因此, 为了找到多个帕累托最优解, 设计特殊的搜索机制和环境选择策略至关重要, 对多模态优化的研究具有很大的理论研究和实际应用意义^[4]。

2 相关工作

这一部分主要介绍 MMO 的相关定义, 然后给出差分进化算法的基本框架, 分析求解 MMOP 的困难和选择 DE 的原因。

2.1 MMO 的相关定义

一般来说, PS 是指全局 PS, 可以定义为: 如果在整个可行区域中没有任何其他解支配其元素, 则该解集称为全局 PS^[5]。类似于多模态单目标优化中的局部峰值, MMO 中也存在局部 PS。如果解集的任何元素都不被其局部邻居支配, 则该解集称为局部 PS。

给定一个多目标优化问题，若满足下列条件之一，则属于 MMOP^[6]:

- (1) 至少有一个局部 PS;
- (2) 它有一个以上的全局 PS。

注意，在上面的定义中，全局 PS 不属于局部 PS。因此，如果一个问题有一个局部 PS，那么它必须至少有一个全局 PS。在极端情况下，局部 PS 可能只有一个解。

一般来说，有两种类型的 MMOPs。第一种类型只有全局 PSs，第二种类型既有全局 PSs 又有局部 PSs^[7]。该论文只研究了第一类 MMOP。

2.2 差分进化

DE 算法^[8-10]是一种简单但非常有效的进化算法。类似于其他基于种群的随机算法，差分进化的过程主要包括四个步骤：种群初始化、变异、重组和环境选择。在变异步骤中，有几种不同的 DE 变异策略，如 DE/rand/1、DE/best/1、DE/best/2 和 DE/rand/2。由于需要获得多个解集，所以最好具有较强的探索能力。因此，本文采用 DE/rand/2。差分向量生成如下：

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

其中 V_i 表示差分向量， r_1, r_2, r_3, r_4 和 r_5 是不相等的整数。F 是用于缩放差向量的比例因子。该论文提出了一种基于欧氏距离的小生境方法^[11-13]被嵌入到 DE 算法中， r_1, r_2, r_3, r_4 和 r_5 根据它们在决策或目标空间中的拥挤距离在相应个体的邻居中被选择。详细的选择方法在第 3 节中介绍。之所以选择 DE，是因为它简单且在增强种群多样性方面相当有效^[11,14-15]。在 MMO，提高种群多样性意义重大。因此，采用改进 DE/rand/2 操作的差分进化算法求解多目标优化问题。

解决 MMOPs^[6,16]有几个挑战。首先，一旦发现一个 PSs，收敛强度将显著降低。这是因为当获得一个 PSs 时，就找到了整个 PF。除了仅在其他 PSs 上的解之外，其他解都由所获得的 PSs 支配。所以收敛强度会降低。第二，在环境选择中很难同时存在多个等价的帕累托最优解。这些等价的 Pareto 最优解映射到目标空间中的同一点。如果以目标空间中的拥挤距离作为环境选择标准，它们会拥挤得无法生存。

3 本文方法

3.1 MMODE_ICD 框架

MMODE_ICD 算法流程如图 1 所示。首先，在搜索空间中初始化群体。然后，自适应地选择产生差异向量的个体，根据式(1)对种群进行变异。第三，子代与亲代组合成 P_combination。根据非支配等级和改进的拥挤距离 (ICD) 和特殊拥挤距离 (SCD) 从 P_combination 中选择下一代种群。具有小的非支配等级值和大的 SCD 的个体被选择的概率更高。在选择过程中，只有特定的比例个体被选入最高的帕累托等级，从而为其他潜在的解决方案提供了机会。自适应个体选择、改进的拥挤距离和比例选择的细节在以下文本中描述。

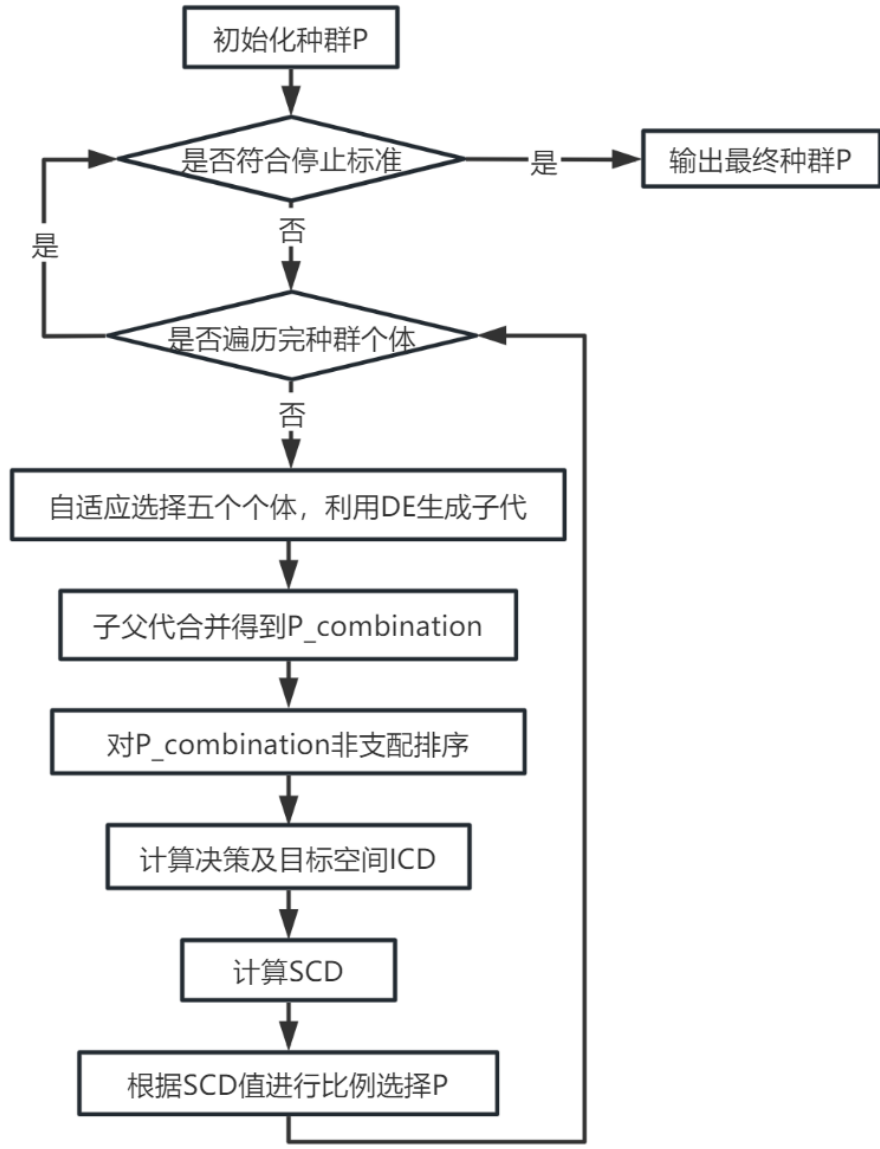


图 1: MMODE_ICD 流程图

3.2 利用自适应方法选择个体生成差异向量

为了提高决策空间和目标空间的多样性，使用以下三种方法自适应地选择产生差异向量的个体。

(1) 在总体中随机选取五个邻居；

(2) 根据决策空间中当前个体与整个种群中剩余解之间的欧氏距离，选择一定数量的邻居。然后，在一定数量的邻居中随机选择五个邻居，选择决策空间中拥挤距离最大的一个作为 x_1 ，其余为 x_2, x_3, x_4, x_5 ；

(3) 根据目标空间中当前个体与整个种群中剩余解之间的欧氏距离，选择一定数量的邻居。然后，在一定数量的邻居中随机选择五个邻居，选择目标空间中拥挤距离最大的一个作为 x_1 ，其余为 x_2, x_3, x_4, x_5 。

使用第一、二、三种方法概率分为 p_1, p_2, p_3 ，其中 $p_1 = 1 - (G_c - 1) / \text{Max_gen}$, $p_2 = (1 - p_1) / 2$, $p_3 = 1 - p_1 - p_2$ 。随着当前世代数 G_c 从 1 增加到 Max_gen ， p_1 从 1 减少到 0，而 p_2 和 p_3 从 0 增加到 0.5。

以上述方式选择这些个体的原因是自适应地平衡了探索和利用能力，并且提高了决策和目标空间的多样性。一方面，在进化初期随机选择整个种群中概率较大的个体，可以增强探索能力，避免陷入

局部最优区域。在进化后期，选择当前个体的邻居可以增强利用能力。另一方面，在决策空间中选择邻居，选择决策空间中拥挤距离最大的邻居作为基向量，可以提高决策空间的多样性。另一方面，在目标空间中选择邻居，选择目标空间中拥挤距离最大的一个作为基向量，可以提高目标空间的多样性。

3.3 考虑 ICD 中已选择个体

为了提高种群的多样性，拥挤距离被用来衡量种群中的拥挤程度。传统的拥挤距离可以在一定程度上反映个体的拥挤程度。然而，它有一个明显的缺点。单独计算每个帕累托等级的拥挤距离并不能反映真实的拥挤程度，应考虑已经选择的所有个体。

在目标空间中，拥挤距离是利用每个维度的上邻和下邻来计算的，而在决策空间中，拥挤距离是利用与邻的加权欧氏距离的平均值来计算的。设 $CD_{i,x}$ 表示决策空间中第 i 个个体的拥挤距离。那么 $CD_{i,x}$ 的计算如下。

$$CD_{i,x} = \sum_{j=1}^{k_1} (k_1 - j + 1) d_{i,j} \quad (2)$$

其中 $d_{i,j}$ 表示整个种群中第 i 个个体到其邻近的第 j 个个体的欧氏距离，在它的邻域中有 k_1 个邻居。

3.4 Pareto 等级嵌入 SCD

在文献^[17]中，提出了特殊拥挤距离 (SCD)。第 i 个个体的 SCD 定义为：

$$SCD_i = \begin{cases} \max(CD_{i,x}, CD_{i,f}), & CD_{i,x} > CD_{avg,x} \text{ or } CD_{i,f} > CD_{avg,f} \\ \min(CD_{i,x}, CD_{i,f}), & \text{othersize} \end{cases} \quad (3)$$

其中 $CD_{i,x}$ 为决策空间中的拥挤距离， $CD_{i,f}$ 为目标空间中的拥挤距离， $CD_{avg,x}$ 为决策空间中的平均拥挤距离， $CD_{avg,f}$ 为目标空间中的平均拥挤距离。然而，这种 SCD 计算方法平等地处理所有个体。最后几个非主导队列的个体即使在决策空间中具有较好的多样性也不会被选中，因为他们离 PF 较远，对于这些个体来说，决策空间中的多样性比目标空间中的多样性更重要。其原因是提高决策空间的多样性可以扩大搜索范围，增加找到更多帕累托最优解的概率。为了达到这个目标，第 i 个个体的 SCD 被修改为：

$$SCD_i = \begin{cases} \max(CD_{i,x}, \frac{CD_{i,f}}{Rank}), & CD_{i,x} > CD_{avg,x} \text{ or } CD_{i,f} > CD_{avg,f} \\ \min(CD_{i,x}, CD_{i,f}), & \text{othersize} \end{cases} \quad (4)$$

其中 Rank 表示第 i 个个体的非主导 Rank 值。

3.5 比例选择

该环境选择方法只选择一定比例的 Pareto 等级排在前面的个体，所选比例随代数的变化而变化，因为得到的 PF 一代比一代比完整。因此，没有必要挑选等级排在最后的个体，在后期阶段主要任务是提高种群多样性。它们的关系如图 2 和式 (5) 所示。

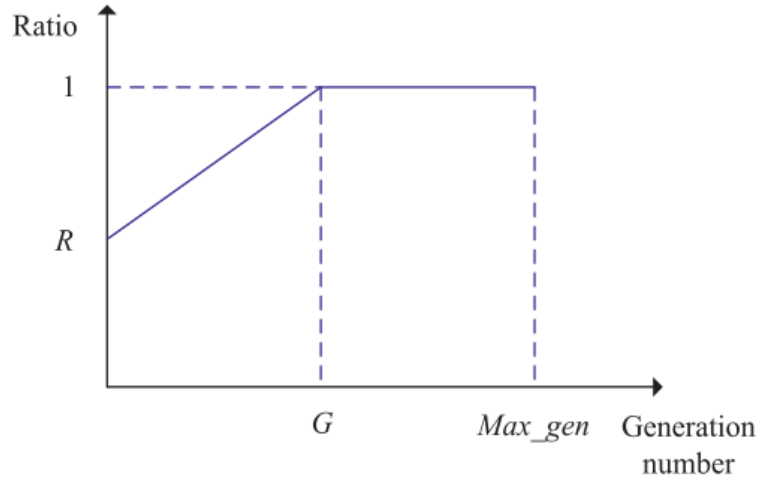


图 2: 比例与世代数的关系图

$$Ratio = \begin{cases} R + \frac{1-R}{G}(G_c - 1), & 1 < G_c < G \\ 1, & G < G_c < Max_gen \end{cases} \quad (5)$$

其中 G_c 为当前代数, Max_gen 为最大代数。

4 复现细节

4.1 与已有开源代码对比

这篇论文的代码是开源的, 因此在作者提供的这个代码的基础上进行了一些改进, 所做的工作主要包括以下几个部分。

该论文一共使用 22 个测试集, 但是 MMF10, MMF11, MMF12, MMF13, MMF15, MMF15_a 这六个测试集不仅有全局 PS 和 PF, 还有局部 PS 和 PF。论文提出的算法只考虑全局 PS, 即上文提到的第一类问题, 因此, 在导入数据时只选择全局 PS 和 PF, 而不是将所有数据导入进行计算。

虽然论文中提出使用自适应的方法选择个体生成差异向量, 但是在作者提供的代码中并没有使用自适应的方法, 而是使用随机锦标赛方法。因此, 对这部分进行复现。

在计算决策空间中某个个体的拥挤度距离时, 利用与该个体邻居的加权欧氏距离的平均值来计算, 并且设置领域中个体邻居数为 $k_1 = 0.2 * popsize$, 尝试使用一种新的方法来计算 k_1 。

绘制了各评价指标图以及种群在进化过程中向 PS 和 PF 的演化过程图。

4.2 创新点

论文所提算法在决策空间表现较差, 并不能找到所有 PS, 解易聚集在某些 PS 上, 决策空间多样性不足。在计算决策空间个体拥挤距离时, 根据已选择个体在决策空间中的值来计算某个个体的邻居, 而不是使用固定的邻居数。

如果决策空间中 x_i 和 x_n 的距离小于 k_1 , 则 x_n 称为解 x_i 的邻居, 其计算如下:

$$k_1 = \eta \left(\prod_{m=1}^M (x_m^{max} - x_m^{min}) \right) \frac{1}{M} \quad (6)$$

其中 M 为决策变量个数, x_m^{max} 和 x_m^{min} 分别为当前演化过程中第 m 个决策变量的最大值和最小值。 k_1 的第二部分用于评估决策空间的平均距离。 η 是一个经验参数, 用于控制邻居的平均长度, 根据^[18]设

置为 0.2。

5 实验结果分析

5.1 实验设置

论文中使用了 22 个测试集，因此在本次改进算法的实验中也使用这 22 个测试集，在这些测试集中，有些具有相似的特点，如 PS 和 PF 形状为凹或者凸、是否连续，变量维度等。其他参数与论文中设置一样，种群大小被设置为 $100 * n_{var}$ ，最大适应度评估被设置为 $5000 * n_{var}$ ，其中 n_{var} 表示决策变量的数量。(1)中的 F 设置为 0.5，Crossover Rate(CR) 设置为 0.5，所有实验都独立进行 21 次。使用 $1/PSP^{[6]}$ 和 $1/HV^{[19]}$ 两个评价指标评估它们的性能。其中， $1/PSP$ 可以度量决策空间中的性能， $1/HV$ 用于度量目标空间中的性能。对于这两个指标，值越小意味着性能越好。

因为论文有提供源代码，因此改进的算法不仅与论文中所述的结果进行比较，还与论文提供的源代码进行对比。

5.2 评价指标值对比

算法在各测试集上得到的 rHV 结果如表 1 所示。在 MMF2 和 MMF3 上改进算法得到的 rHV 比论文以及源代码中得到的值要好 0.01、0.02 左右，在 MMF10、MMF14、MMF1_e 和 MMF14_a 上，论文中的结果比源代码以及改进结果好 0.01、0.02 左右，而在其他测试集上得到的值比较接近，因此他们在目标空间上的表现结果差不多。

表 1: 算法在各测试集上得到的 rHV 值

测试集	论文结果	源代码结果	改进结果
MMF1	1.1458	1.1462	1.1458
MMF2	1.1765	1.1538	1.1504
MMF3	1.1672	1.1516	1.1492
MMF4	1.8522	1.8533	1.8532
MMF5	1.1461	1.1460	1.1463
MMF6	1.1456	1.1461	1.1461
MMF7	1.1453	1.1462	1.1457
MMF8	2.3764	2.3760	2.3759
MMF9	0.1032	0.1033	0.1032
MMF10	0.0784	0.0816	0.0807
MMF11	0.0689	0.0689	0.0689
MMF12	0.6355	0.6360	0.6364
MMF13	0.0542	0.0542	0.0542
MMF14	0.3131	0.3320	0.3354
MMF15	0.2211	0.2257	0.2261
MMF1_z	1.1455	1.1461	1.1456
MMF1_e	1.1482	1.1692	1.1551
MMF14_a	0.3181	0.3245	0.3306
MMF15_a	0.2242	0.2251	0.2254
SYM-PARTsimple	0.0600	0.0601	0.0601
SYM-PARTrotated	0.0601	0.0601	0.0601
Omni-test	0.0189	0.0192	0.0193

算法在各测试集上得到的 rPSP 结果如表 2 所示。因为 MMF10、MMF11、MMF12、MMF13、MMF15 以及 MMF15_a 这些测试集既有全局 PS、PF 又有局部 PS、PF，但是在改进的实验中，只导入全局 PS、

PF，因此其得到的值要小很多。由表 2 可知，改进算法得到的结果在 MMF2-3、MMF10-13、MMF15、MMF1_e 以及 MMF15_a 中得到的结果要好 0.01、0.02 左右。论文中的结果在 MMF1、MMF4-6、MMF8、MMF1_z、SYM-PARTsimple、SYM-PARTrotated 以及 Omni-test 中得到的结果要好 0.01、0.02 左右。

表 2: 算法在各测试集上得到的 rPSP 值

测试集	论文结果	源代码结果	改进结果
MMF1	0.0493	0.0552	0.0604
MMF2	0.0247	0.0115	0.0106
MMF3	0.0209	0.0089	0.0088
MMF4	0.0257	0.0329	0.0327
MMF5	0.0853	0.1470	0.1085
MMF6	0.0713	0.0973	0.0820
MMF7	0.0263	0.0260	0.0273
MMF8	0.1303	0.2570	0.1445
MMF9	0.0047	0.0061	0.0061
MMF10	0.2011	0.1362	0.0607
MMF11	0.2521	0.2501	0.0041
MMF12	0.2474	0.2451	0.0016
MMF13	0.5370	0.5328	0.0211
MMF14	0.0422	0.0453	0.0463
MMF15	0.2664	0.2570	0.0353
MMF1_z	0.0387	0.0442	0.0481
MMF1_e	2.6551	1.2215	1.6610
MMF14_a	0.0580	0.0547	0.0551
MMF15_a	0.2674	0.2568	0.0382
SYM-PARTsimple	0.0427	0.8614	0.2012
SYM-PARTrotated	0.0892	0.7012	0.2063
Omni-test	0.0512	0.3203	0.1910

本次研究主要是针对论文中提出的算法在决策空间表现较差问题进行改进，由以上分析可知，改进算法在 rHV 上得到的值与论文测试结果很接近，在将近一半的 MMF 测试集上得到的 rPSP 值较优，即改进算法在保持目标空间多样性的条件下，能在一定程度上增强决策空间的多样性以及收敛性。

5.3 PS 和 PF 分布图对比

二维空间以 MMF3 测试集为例，各算法最终得到的 PS 和 PF 分布图如图 3（图 4为其局部放大图）、图 5（图 6为其局部放大图）所示。从图中可以看出，论文提供的源代码与改进代码得到的 PF 与真实 PF 很接近，且分布较均匀，具有较好的收敛性和多样性。在 MMF3 测试集上得到的种群较均匀的分布在真实 PS 上，但是在其他测试集上，种群在决策空间多样性还有待提高。

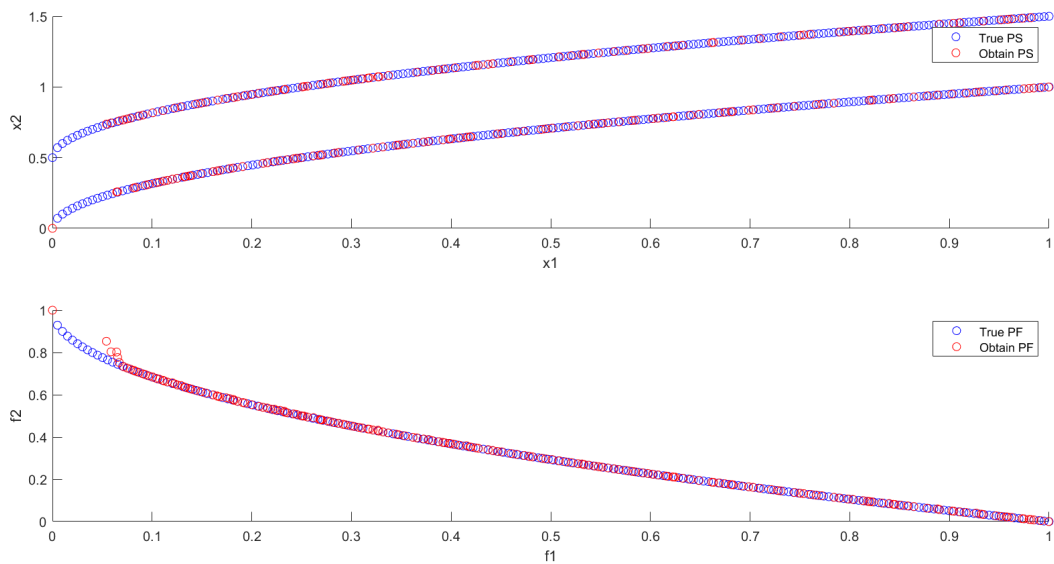


图 3: 源代码在 MMF3 上运行得到的 PS、PF 分布图

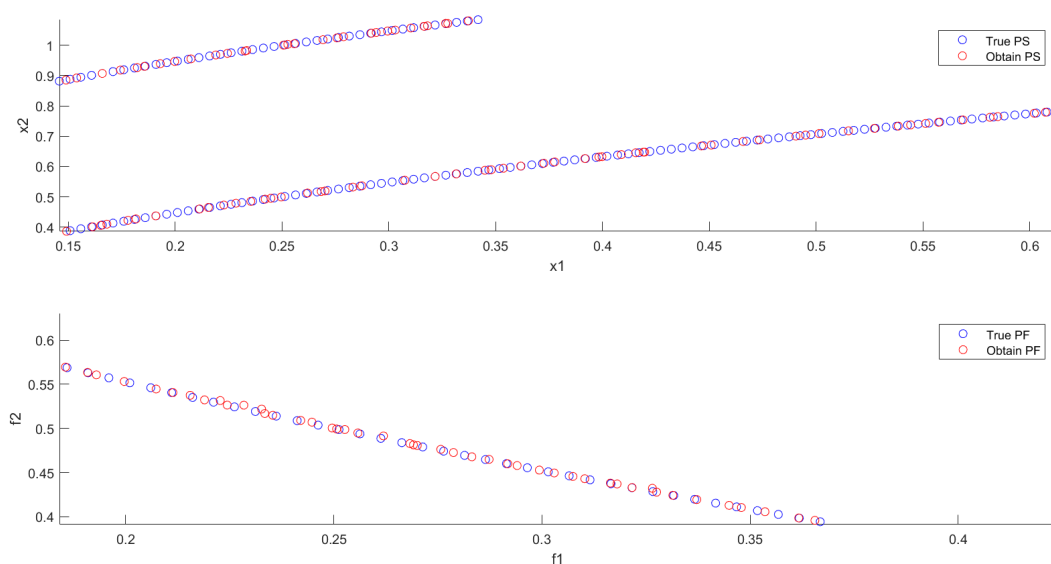


图 4: 源代码在 MMF3 上运行得到的 PS、PF 局部放大分布图

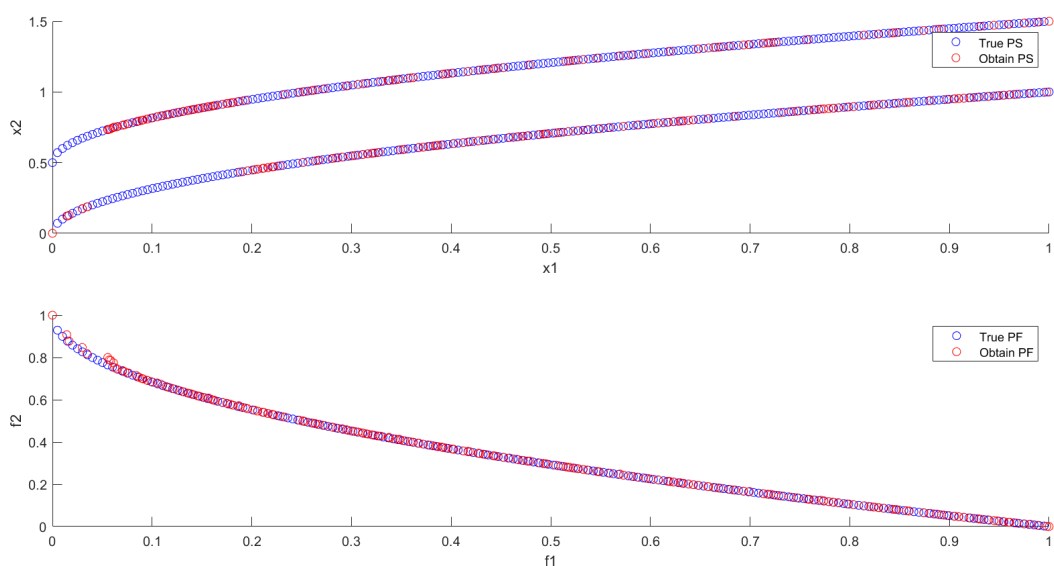


图 5: 改进代码在 MMF3 上运行得到的 PS、PF 分布图

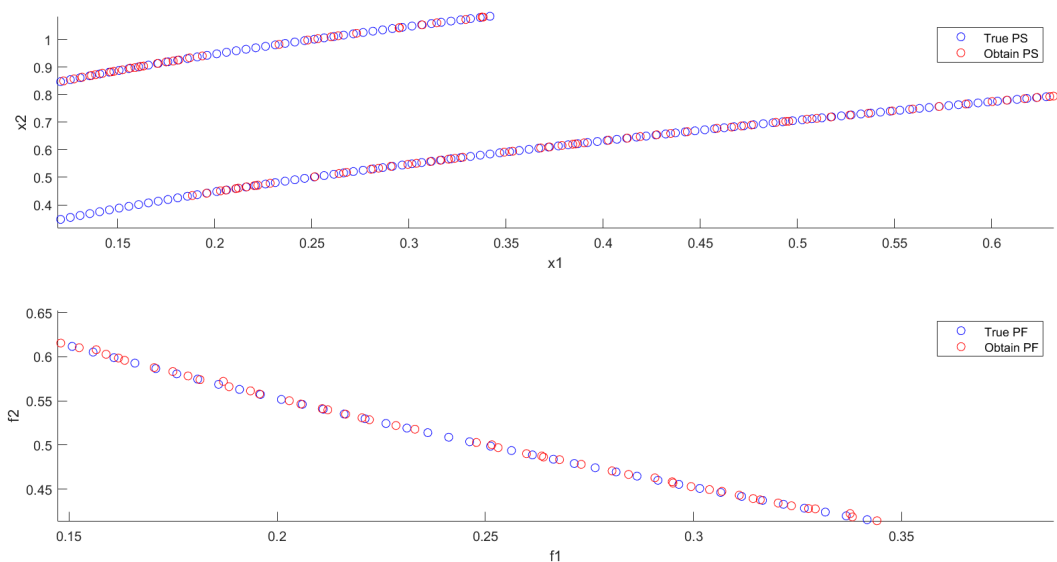


图 6: 改进代码在 MMF3 上运行得到的 PS、PF 局部放大分布图

三维空间以 MMF13 测试集为例，MMF13 测试集的决策空间是三维的，目标空间为二维，各算法最终得到的 PS 和 PF 分布图如图 7（图 8 为其局部放大图）、图 9（图 10 为其局部放大图）所示。图 7 最上层蓝色圆圈代表的是局部 PS 和 PF，该论文在导入数据时并没有进行筛选，且在计算各评价指标时也考虑了局部 PS 和 PF，但是论文所提出的算法只考虑第一类问题，即全局 PS 和 PF。因此，论文中计算各评价指标时并不客观。针对该问题，改进算法在导入数据时只选择全局 PS 和 PF。

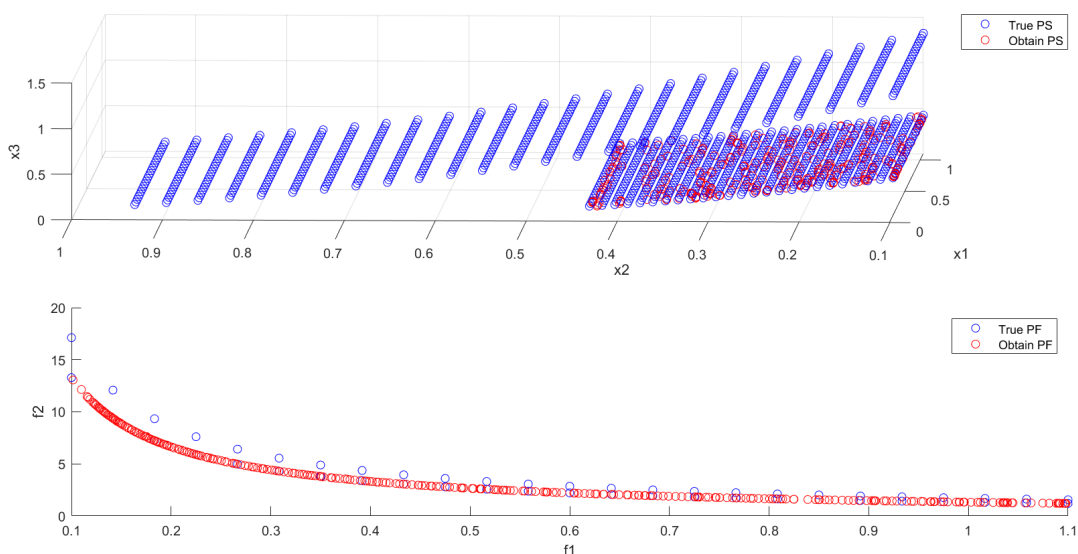


图 7: 源代码在 MMF13 上运行得到的 PS、PF 分布图

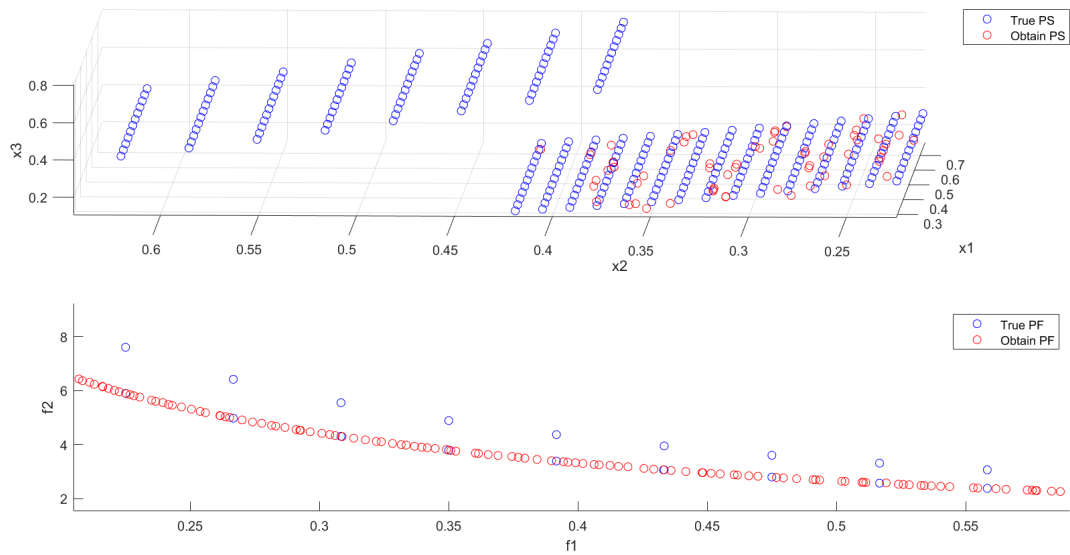


图 8: 源代码在 MMF13 上运行得到的 PS、PF 局部放大分布图

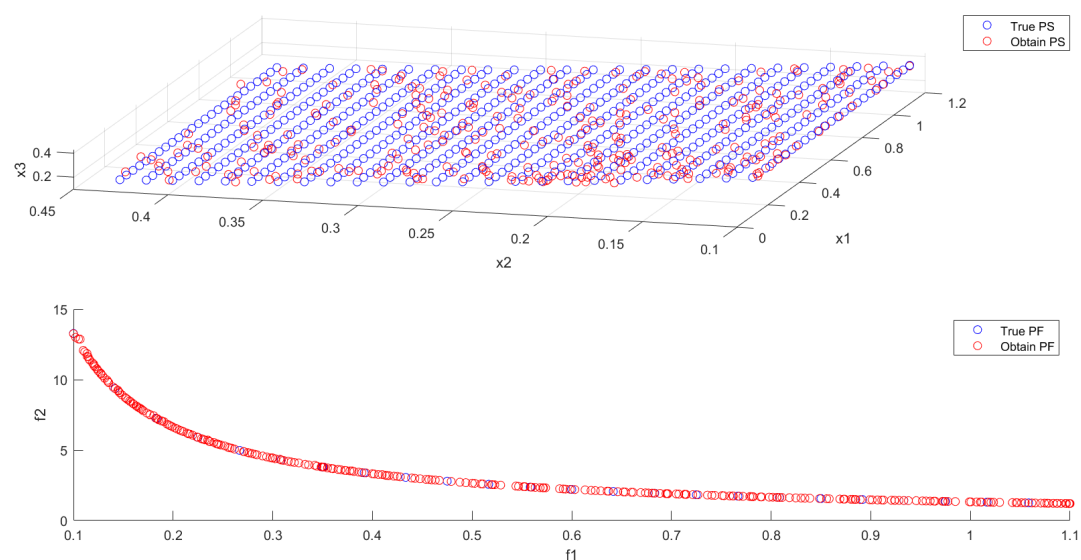


图 9: 改进代码在 MMF13 上运行得到的 PS、PF 分布图

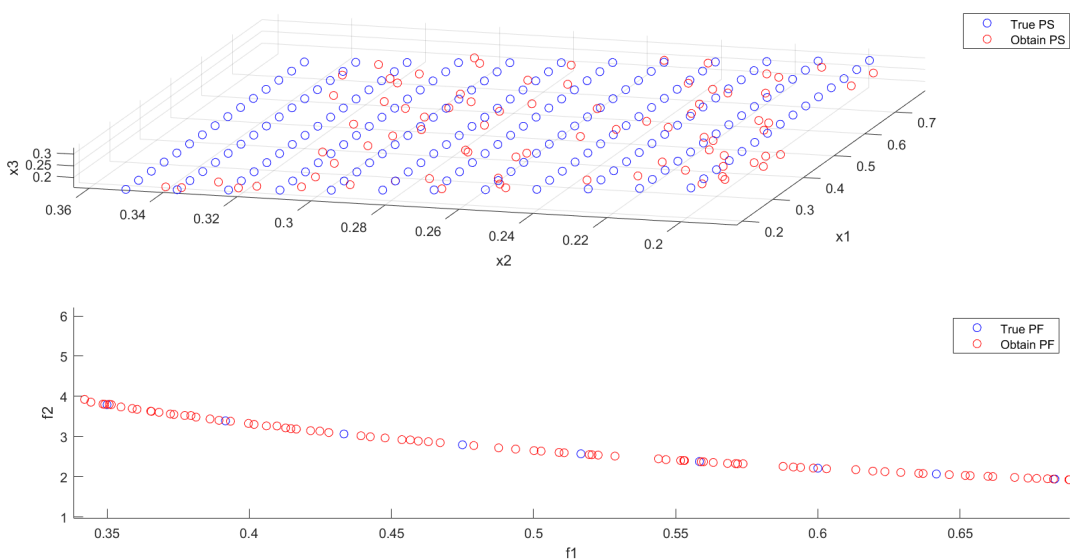


图 10: 改进代码在 MMF13 上运行得到的 PS、PF 局部放大分布图

6 总结与展望

该论文提出了一种改进的差分进化算法来求解多模态多目标优化问题。这篇论文仅考虑第一类 MMOP, 因此在导入测试集数据时只导入全局 PS 和 PF。在改进的方法中, 考虑决策空间和目标空间的拥挤距离, 自适应地选择产生差分向量的个体, 并对该部分进行复现。设计了一种改进的拥挤距离计算方法来平衡决策空间和目标空间的分布, 拥挤距离邻居使用公式(6)计算, 而不是使用论文中固定的参数 k_1 , 因此更能反映真实的拥挤程度。此外, 非支配排序值被嵌入到拥挤距离计算中, 使得决策空间中的拥挤度在随后的排序中被赋予优先权。在环境选择中, 只有一定比例的个体被适应性地选择, 从而给潜在的优势个体机会。最后, 绘制各评价指标图以及种群在进化过程中向 PS 和 PF 的演化过程图。

但该算法在决策空间的性能还需要进一步提高。其次, 它的计算复杂度相对较高。第三, 不容易为不同种类的问题设置参数。在多模态多目标优化方面还有许多工作要做。首先, 可以研究带有局部和全局 PS 的 MMOP。第二, 研究依赖于群体当前状态的参数自适应机制。

参考文献

- [1] DEVAURS D, SIMÉON T, CORTÉS J. Optimal Path Planning in Complex Cost Spaces With Sampling-Based Algorithms[J]. IEEE Transactions on Automation Science and Engineering, 2016, 13: 415-424.
- [2] 王凌, 邓瑾, 王圣尧. 分布式车间调度优化算法研究综述[J]. 控制与决策, 2016, 31(1): 1.
- [3] 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: 387-402.
- [4] 岳彩通, 梁静, 瞿博阳, 等. 多模态多目标优化综述[J]. 控制与决策, 2021, 36(11): 2577.
- [5] DEB K. Multi-objective genetic algorithms: problem difficulties and construction of test problems[J]. Evol. Comput, 1999, 7(3): 205-230.
- [6] YUE C, QU B, YU K, et al. A novel scalable test problem suite for multimodal multiobjective optimization[J]. Swarm and Evolutionary Computation, 2019, 48: 62-71.
- [7] LIANG J, SUGANTHAN P, QU B, et al. Definitions and Evaluation Criteria for the CEC 2020 Special Session on Multimodal Multiobjective Optimization[J]. Zhengzhou University and Nanyang Technological University, 2020.
- [8] DAS S, SUGANTHAN P N. Differential Evolution: A Survey of the State-of-the-Art[J]. IEEE Transactions on Evolutionary Computation, 2011, 15(1): 4-31.
- [9] QIN A K, HUANG V L, SUGANTHAN P N. Differential Evolution Algorithm With Strategy Adaptation for Global Numerical Optimization[J]. IEEE Transactions on Evolutionary Computation, 2009, 13(2): 398-417.
- [10] KARABOGA D, AKAY B. A comparative study of Artificial Bee Colony algorithm[J]. Applied Mathematics and Computation, 2009, 214(1): 108-132.

- [11] QU B Y, SUGANTHAN P N, LIANG J J. Differential Evolution With Neighborhood Mutation for Multimodal Optimization[J]. IEEE Transactions on Evolutionary Computation, 2012, 16(5): 601-614.
- [12] SUGANTHAN P. Particle swarm optimiser with neighbourhood operator[J]., 1999, 3: 1958-1962 Vol. 3.
- [13] QU B Y, 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.
- [14] WANG Y, YANG Z, GUO Y, et al. A novel multi-objective competitive swarm optimization algorithm for multi-modal multi objective problems[J]., 2019: 271-278.
- [15] BASAK A, DAS S, TAN K C. Multimodal Optimization Using a Biobjective Differential Evolution Algorithm Enhanced With Mean Distance-Based Selection[J]. IEEE Transactions on Evolutionary Computation, 2013, 17(5): 666-685.
- [16] DEB K, TIWARI S. Omni-optimizer: A Procedure for Single and Multi-objective Optimization[J]., 2005.
- [17] YUE C, QU B, LIANG 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.
- [18] LI W, YAO X, ZHANG T, et al. Hierarchy Ranking Method for Multimodal Multi-objective Optimization with Local Pareto Fronts[J]. IEEE Transactions on Evolutionary Computation, 2022: 1-1.
- [19] ZITZLER E, THIELE L, LAUMANN M, et al. Performance assessment of multiobjective optimizers: an analysis and review[J]. IEEE Trans. Evol. Comput., 2003, 7: 117-132.