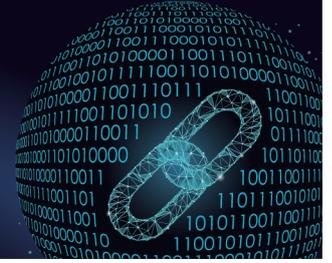




中国科技期刊卓越行动计划项目入选期刊

# 控制与决策

CONTROL AND DECISION



## 基于自适应聚合距离的多目标进化算法

曾亮, 曾维钧, 李燕燕, 全睿, 王珊珊

引用本文:

曾亮, 曾维钧, 李燕燕, 全睿, 王珊珊. 基于自适应聚合距离的多目标进化算法[J]. 控制与决策, 2024, 39(4): 1113–1122.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2022.1342>

## 您可能感兴趣的其他文章

### Articles you may be interested in

#### 一种求解约束多目标问题的协作进化算法

A collaborative evolutionary algorithm for solving constrained multi-objective problems

控制与决策. 2021, 36(11): 2656–2664 <https://doi.org/10.13195/j.kzyjc.2020.0791>

#### 基于向量角分解的高维多目标进化算法

Many-objective evolutionary algorithm based on vector angle decomposition

控制与决策. 2021, 36(3): 761–768 <https://doi.org/10.13195/j.kzyjc.2019.0925>

#### 基于弱关联的自适应高维多目标进化算法

A weak association-based adaptive evolutionary algorithm for many-objective optimization

控制与决策. 2021, 36(8): 1804–1814 <https://doi.org/10.13195/j.kzyjc.2019.1723>

#### 基于多种群分解预测的动态多目标引力搜索算法

Dynamic multi-objective gravitational searching algorithm based on multi-population decomposition prediction

控制与决策. 2021, 36(12): 2910–2918 <https://doi.org/10.13195/j.kzyjc.2020.1002>

#### 基于改进多目标优化算法的分布式数据中心负载调度

Multi-objective optimization of energy and performance management in distributed data centers

控制与决策. 2021, 36(1): 159–165 <https://doi.org/10.13195/j.kzyjc.2019.0702>

# 基于自适应聚合距离的多目标进化算法

曾亮<sup>1,2,3</sup>, 曾维钧<sup>1,2</sup>, 李燕燕<sup>1,2</sup>, 全睿<sup>1,2</sup>, 王珊珊<sup>1,2†</sup>

(1. 湖北工业大学 电气与电子工程学院, 武汉 430068; 2. 太阳能高效利用及储能运行控制  
湖北省重点实验室, 武汉 430068; 3. 襄阳湖北工业大学产业研究院, 湖北 襄阳 441100)

**摘要:** 随着目标数的增多,种群收敛性与分布性的冲突愈加激烈,传统的多目标进化算法的选择算子难以平衡种群的收敛性与分布性. 对此,提出一种基于自适应聚合距离的多目标进化算法. 首先,采用参考点支配关系替代原有的Pareto支配关系,以增加选择压力,加强收敛性;其次,提出自适应聚合距离,通过动态变化的惩罚参数来自适应调整收敛性与分布性的比例;最后,设计一种带有淘汰算子的方法以改进小生境选择策略,根据自适应聚合距离的大小进行选择和淘汰操作. 为验证算法的可行性,将所提出算法在测试问题上与其他4种优秀的多目标进化算法进行比较,并应用于两个实际应用中,仿真结果表明,所提出算法的综合性能更优,能有效平衡种群的收敛性与分布性.

**关键词:** 多目标进化算法; 参考点支配; 自适应聚合距离; 淘汰算子; 收敛性与分布性

**中图分类号:** TP18 **文献标志码:** A

**DOI:** 10.13195/j.kzyjc.2022.1342

**引用格式:** 曾亮,曾维钧,李燕燕,等. 基于自适应聚合距离的多目标进化算法[J]. 控制与决策, 2024, 39(4): 1113-1122.

## A multi-objective evolutionary algorithm based on adaptive aggregation distance

ZENG Liang<sup>1,2,3</sup>, ZENG Wei-jun<sup>1,2</sup>, LI Yan-yan<sup>1,2</sup>, QUAN Rui<sup>1,2</sup>, WANG Shan-shan<sup>1,2†</sup>

(1. School of Electrical and Electronic Engineering, Hubei University of Technology, Wuhan 430068, China; 2. Hubei Key Laboratory for High-efficiency Utilization of Solar Energy and Operation Control of Energy Storage System, Wuhan 430068, China; 3. Xiangyang Industrial Institute of Hubei University of Technology, Xiangyang 441100, China)

**Abstract:** The conflict between convergence and distribution of population will intensify when the dimensionality of objective increases. The selection operator of the traditional multi-objective evolutionary algorithm is difficult to balance the convergence and distribution of the population. To solve this problem, a multi-objective evolutionary algorithm based on adaptive aggregation distance (MOEA-AAD) is proposed. Firstly, the reference point dominance relation is used to replace the original Pareto dominance relation to increase the selection pressure and strengthen the convergence. Then, an adaptive aggregation distance is proposed to adaptively adjust the ratio of convergence and distribution through dynamically changing penalty parameters. Finally, a method with a deletion operator is designed to improve the niche selection strategy, and the selection and elimination operation are carried out according to the value of the adaptive aggregation distance. In order to prove the optimization performance of the proposed algorithm, it is compared with four other excellent multi-objective evolutionary algorithms on the test problem and applied to two real-world applications. The simulation results show that the proposed algorithm has better comprehensive performance, which can effectively balance the convergence and distribution of the population.

**Keywords:** multi-objective evolutionary algorithm; reference point dominance; adaptive aggregation distance; deletion operator; convergence and distribution

收稿日期: 2022-07-26; 录用日期: 2022-12-08.

基金项目: 国家自然科学基金项目(51977061); 湖北省重点研发计划项目(2020BAB114); 湖北省教育厅科学研究计划重点项目(D20211402); 襄阳湖北工业大学产业研究院2022年度项目(XYYJ2022C04).

责任编委: 巩敦卫.

†通讯作者. E-mail: wangshanshan@hbut.edu.cn.

\*本文附带电子附录文件,可登录本刊官网该文“资源附件”区自行下载阅览.

## 0 引言

社会生产中存在许多多目标优化问题(MOPs),如流水车间调度<sup>[1]</sup>和物资分配<sup>[2]</sup>等,各目标之间往往存在相互冲突的部分,且得到的是一组折衷解而不是能同时使所有目标取得最优的某一个解<sup>[3]</sup>.当目标的个数超过3时,称其为高维多目标问题(MaOPs).

虽然一些性能优越的多目标进化算法(MOEAs)如NSGA-II<sup>[4]</sup>、MOEA-PPF<sup>[5]</sup>能很好地处理MOPs,但在处理高维多目标优化问题时效果不佳,主要原因是随着目标函数的增加,相互冲突的目标数量较多,非支配个体在种群中所占的比重呈指数增长,导致算法计算高维多目标问题时的计算复杂度也呈现急剧上升的趋势,计算将会变得缓慢且计算精度得不到保证,算法的性能下降严重.基于Pareto支配关系的进化算法会失去作用,解集难以收敛且分布性难以维持.

为解决以上难题,研究人员从不同切入点对多目标进化算法进行了深入研讨,按照它们的特点大致归类为4个类型:第1类是基于松弛支配关系的多目标进化算法<sup>[6-8]</sup>,但松弛支配关系高度依赖参数的设置,且实际应用较少;第2类是基于分解的多目标进化算法,其核心是对参考向量和分解方法的设计,如MOEA/D<sup>[9]</sup>和NSGA-III<sup>[10]</sup>,后来不少学者基于MOEA/D框架进行研究并获得了理想的效果<sup>[11-12]</sup>,虽然该类算法可以提高运行速度,但缺乏对权重向量的考虑;第3类是基于性能指标的多目标进化算法,如CDR-MOEA<sup>[13]</sup>和MaOEA-IBP<sup>[14]</sup>,这类算法的计算成本高,难以快速验证结果;第4类是混合型的多目标进化算法,利用两种或更多的方法的优势实现优化,如SDIEA<sup>[15]</sup>和H-RVEA<sup>[16]</sup>.

基于参考点的NSGA-III算法忽略了趋近Pareto前沿的收敛性,面对高维问题时缺乏足够的选择压力,分布性的维护能力也明显不足<sup>[17-18]</sup>.因此,为了平衡算法的收敛性与分布性,本文沿袭NSGA-III的框架,提出基于自适应聚合距离的多目标进化算法(MOEA-AAD).其中,针对Pareto支配机制的选择压力低下的问题,采用一种参考点支配方式对种群个体进行等级划分,以提高收敛能力.在环境选择中引入自适应聚合距离动态调整惩罚参数,并设计小生境选择与淘汰策略,根据自适应聚合距离选择优良个体.采用多种测试函数以及实际应用问题对算法进行验证,并与多个优秀的多目标进化算法进行对比,结果表明MOEA-AAD算法具有更好的综合性能.

## 1 相关研究

### 1.1 多目标优化问题的基本定义

一般情况下,多目标优化问题的数学表达如下:

$$\begin{aligned} \min F(x) &= (f_1(x), f_2(x), \dots, f_M(x))^T; \\ \text{s.t. } x &\in \Omega. \end{aligned} \quad (1)$$

其中: $x = (x_1, x_2, \dots, x_n)^T$ 为决策向量, $\Omega$ 为决策空间, $F(x) \rightarrow R^M$ 为 $m$ 维目标函数, $R^M$ 为目标空间.

### 1.2 PBI聚合函数

边界惩罚的交叉法 (penalty-based boundary intersection, PBI) 广泛应用于多目标优化领域,设 $Z^j$ 为参考向量, $F(x)$ 为目标向量,PBI聚合函数为

$$d(x) = d_1(x) + \theta d_2(x). \quad (2)$$

其中

$$d_1(x) = \frac{\|F(x)Z^j\|}{\|Z^j\|}, \quad (3)$$

$$d_2(x) = \left\| F(x) - d_1 \left( \frac{Z^j}{\|Z^j\|} \right) \right\|, \quad (4)$$

$d_1$ 为投影距离, $d_2$ 为垂直距离, $\theta$ 为惩罚参数.PBI法的实质是调节 $d_1$ 和 $d_2$ ,以平衡收敛性与分布性.

### 1.3 基于参考点的支配关系

参考点支配关系是一种通过整合Pareto支配和基于分解思想的新型支配关系.参考点支配的准则是:对于给定的两个个体 $x$ 和 $y$ ,如果 $x$ 支配 $y$ ,记为 $x \prec_{RP} y$ ,则需满足以下条件之一:

- 1)  $x$  Pareto 支配  $y$ .
- 2)  $x$  和  $y$  的 Pareto 等级相等:
  - ①  $RP(x) = RP(y)$  并且  $d_1(x) < d_1(y)$ ;
  - ②  $RP(x) \neq RP(y)$ ,  $d_1(x) < d_1(y)$  并且  $RP \text{ Density}(x) < RP \text{ Density}(y)$ .

其中: $RP(x)$ 表示与 $x$ 关联的参考点, $RP(y)$ 表示与 $y$ 关联的参考点, $RP \text{ Density}(x)$ 表示与 $RP(x)$ 相关联的个体的数目, $RP \text{ Density}(y)$ 表示与 $RP(y)$ 相关联的个体的数目,即拥挤的程度.关于参考点支配关系的详细介绍可借鉴文献[19].

## 2 基于自适应聚合距离的多目标进化算法

### 2.1 算法的整体框架

为了更好地平衡多目标进化算法的收敛性与分布性,本文提出基于自适应聚合距离的多目标进化算法,具体流程如算法1所示.

**算法1** MOEA-AAD整体框架.

输入: 种群规模 $N$ ,参考点数量 $N_R$ ,目标维数 $M$ ,最大迭代次数 $T_{\max}$ ;

输出: 种群 $P$ .

```

step 1: random initialize ( $N$ );
step 2: reference points generation ( $N_R$ );
step 3:  $t = 1$ ;
step 4: while  $t < T_{\max}$  do
step 5:    $P' = \text{mating} - \text{selection} (P)$ ;
step 6:    $Q = \text{genetic operation} (P')$ ;
step 7:    $S = P \cup Q$ ;
step 8:    $P = \text{environmental selection} (S)$ ;
step 9:    $t = t + 1$ ;
step 10: end while
step 11: return  $P$ .

```

首先,随机生成规模大小为 $N$ 的种群 $P$ ;然后,由父代种群 $P$ 通过遗传操作生成子代种群 $Q$ ,合并到父代以进行环境选择;最后,选出优良个体进入下一代.重复上述操作直到满足终止条件.匹配选择、关联操作及遗传操作详见文献[10],下面重点介绍环境选择及其细节.

## 2.2 环境选择

环境选择如算法2所示.

**算法2** 环境选择.

输入: 种群大小 $N$ , 合并种群 $S$ ;

输出: 种群 $P$ .

```

step 1:  $(F_1, F_2, \dots) = \text{NPD-sort}(S)$  /* NPD-sort( $S$ )为参考点非支配排序,具体参见文献[19]*/;
step 2:  $S_t = \emptyset, i = 1$ ;
step 3: while  $S_t < N$  do
step 4:    $S_t \cup F_i$ ;
step 5:    $i = i + 1$ ;
step 6: end while
step 7: 最后一层为  $F_l = F_i$ ;
step 8: if  $|S_t| = N$  then
step 9:   return  $P$ 
step 10: else
step 11:  normalization( $S$ ) /*归一化操作*/;
step 12: 从临界层中选择的个体  $K = N - |P|$ ;
step 13: 或删除的个体  $Y = F_l - K$ ;
step 14:   $[\pi(s), d(s)] \leftarrow \text{association-operation}$  /*关联操作,  $\pi(s)$ 为个体 $s$ 所关联到距离最近的参考点的索引*/;
step 15:   $d(s) = d_1(s, \pi(s)) + \theta \cdot d_2(s, \pi(s))$  /*自适应聚合距离,详细步骤见2.3.1节*/;
step 16:   $P \leftarrow \text{novel niche strategy}$  /*改进的小生境保留策略,详细步骤见2.3.2节*/;
step 17: end if

```

```

step 18: return  $P$ .

```

首先,利用基于参考点非支配排序法对 $S$ 的个体进行等级划分,再将每一层的个体依序放入临时种群 $S_t$ 内,直至 $S_t \geq N$ .假定 $F_l$ 为临界层,若 $|S_t| = N$ ,则将临时种群 $S_t$ 视作下一代种群 $P$ ;若 $|S_t| > N$ ,则将 $S_t/F_l$ ,即前 $l-1$ 层的个体加入到下一代种群 $P$ 中,剩余的个体基于自适应聚合距离(adaptive aggregated distance, AAD)进行选择,从 $F_l/P$ 中选择 $K = N - |P|$ 个个体保留至下一代中.然后对 $S_t$ 进行归一化操作,在关联操作上,将个体关联到具有最小AAD的参考点上.最后执行改进的小生境保留操作,保留更优的个体.重复改进的小生境保留操作,直至种群 $P$ 的个体数达到 $N$ .

## 2.3 基于自适应聚合距离的精英选择

本文提出一种精英选择策略,与PBI方法类似,通过计算 $d_1$ 和 $d_2$ 两个距离从而得出聚合距离公式.然而大多数方法均采用固定的惩罚因子,忽略了问题的特性和种群进化的实时信息,这种做法会使个体进化出现偏差,进而降低算法的收敛性与分布性.本文引入自适应的聚合距离,在进化过程中自适应地调整惩罚因子,以达到调节收敛性和分布性的作用.另外,为适应高维空间的特性,采用基于小生境的选择与淘汰策略,通过自适应聚合距离来评价个体的质量,从而做出选择或淘汰操作.

### 2.3.1 自适应聚合距离

距离 $d_1$ 越小表示收敛性越好,垂直距离 $d_2$ 越小表示分布性越好.为了使算法在收敛性与分布性方面实现良好折衷,在小生境保留操作中引用一种能同时考虑收敛信息和分布信息的距离函数,采用该距离代替原有个体到参考线的垂直距离.

PBI方法中,参数 $\theta$ 的设置过大或过小都可能导致算法的性能严重下降. $\theta$ 的变化会影响归一化的过程,从而影响算法的搜索行为<sup>[20]</sup>.文献[20]指出,由于归一化的不确定性,基于PBI度量的多目标进化算法采用较小的 $\theta$ 值( $\theta > 0$ )能够避免归一化进程产生较大的敏感率.尽管理论上PBI的参数 $\theta$ 值设置得越小,算法的性能越优异,但其有着最小下限值,而且选择合适的 $\theta$ 需要充分考虑真实前沿的形状,即取决于实际案例.事实上,惩罚参数是平衡收敛性与分布性的一个关键因素.当 $\theta$ 值为5时,能很好地平衡收敛性与分布性,从而使得算法达到最优效果,这也是MOEA/D<sup>[9]</sup>和 $\theta$ -DEA<sup>[21]</sup>中采用 $\theta = 5$ 的原因.

然而,式(2)的缺点是 $\theta$ 在进化过程中由始至终保持不变,这会导致获取的最终解集难以逼近参考方

向. 收敛性和分布性的重要性应随着进化进程而自适应变化: 在进化初期, 由于个体分布较为稀疏, 此时应注重加强收敛性使种群加快逼近PF; 在进化末期, 大部分种群已经收敛到PF上, 需要加强分布性使种群更好地在前沿面上分布开来. 为此, 本文提出一种自适应聚合距离, 其表达式为

$$d(x) = d_1(x) + \theta d_2(x), \quad (5)$$

$$\theta(t, Z_j) = \delta \cdot e^{\frac{M-t}{T_{\max}} \cdot \|Z_j\|}. \quad (6)$$

其中:  $j \in \{1, 2, \dots, N\}$ ,  $M$  为目标个数,  $t$  为当前迭代次数,  $T_{\max}$  为最大迭代次数,  $\delta$  为自定义惩罚因子.

由式(5)和(6)可知, 随着目标个数  $M$  的增大, 每个参考向量关联的子区域也增大, 相对于低维度空间, 则需要一个更大的  $\theta$ , 使得个体靠近其对应的参考方向.

### 2.3.2 基于小生境的选择与淘汰策略

本节采用一种选择与淘汰算子策略, 该算子继承了NSGA-III中小生境保留策略的核心思想. 小生境选择与淘汰操作这两个进程同时进行, 由算法2得出  $K$  和  $Y$  值, 它们分别代表从  $F_l$  层中选择和淘汰的个体数, 且满足  $k + Y = |F_l|$ . 首先, 当  $k < K$  和  $y < Y$  时, 利用选择算子确定具有最小小生境数的参考点集  $J_{\min}$ , 若  $|J_{\min}| > 1$ , 则随机选择一个参考点  $\bar{j}_1 \in j_{\min}$ ; 然后, 找出与该参考点相关联的个体, 并选择具有最小AAD值的个体, 即选择最小  $d$  值的个体, 同时记录  $\rho_{\bar{j}_1}$  加1. 如果仍满足  $k < K$  和  $y < Y$ , 则执行淘汰操作, 删除具有最大AAD值的个体. 经过一次循环后, 选出  $K$  个较优的个体并且淘汰  $Y$  个表现不佳的个体, 以此类推. 一旦不满足循环, 则判断被选出的个体数目情况: 若  $k = K$ , 则表示已完成个体的选择; 否则,  $Y$  个较差的个体被淘汰, 其余个体自动进入下一代. 具体流程见算法3.

**算法3** 基于小生境的选择与淘汰机制.

输入:  $K, Y, S_t, s \in S_t, d(s \in S_t)$ ;

输出: 新一代种群  $P_{t+1}$ .

step 1: 计算小生境的数目

$$\rho_j = \sum_{s \in S_t} (\pi(s) = j) ? 1 : 0;$$

step 2:  $k = 0, y = 0, K + Y = |f_l|$ ;

step 3: while  $k < K$  and  $y < Y$  do

step 4:  $j_{\min} = \{j : \operatorname{argmin}_{j \in Z} \rho_j\}$ ;

step 5:  $\bar{j}_1 = \operatorname{random}(J_{\min})$ ;

step 6:  $I_{\bar{j}_1} = \{s : \pi(s) = \bar{j}_1, s \in S_t\}$ ;

step 7:  $P_{t+1} = P_{t+1} \cup \{s : \operatorname{argmin}_{s \in I_{\bar{j}_1}} d(s)\}$ ;

step 8:  $\rho_{\bar{j}_1} = \rho_{\bar{j}_1} + 1$ ;

step 9:  $k = k + 1$ ;

step 10: while  $k < K$  and  $y < Y$  do

step 11:  $j_{\max} = \{j : \operatorname{argmax}_{j \in Z} \rho_j\}$ ;

step 12:  $\bar{j}_2 = \operatorname{random}(J_{\max})$ ;

step 13:  $I_{\bar{j}_2} = \{s : \pi(s) = \bar{j}_2, s \in S_t\}$ ;

step 14:  $S_t = S_t \setminus \{s : \pi(s) = \bar{j}_2, s \in S_t\}$ ;

step 15:  $\rho_{\bar{j}_2} = \rho_{\bar{j}_2} - 1$ ;

step 16:  $y = y + 1$ ;

step 17: end while

step 18: end while

step 19: if  $k = K$ ;

step 20: continue

step 21: else

step 20: 选择余下的个体到下一代种群  $P_{t+1}$ ;

step 22: end if.

基于自适应聚合距离的精英选择过程如图1所示. 为了更好地说明该过程, 采用两目标问题进行阐述. 图1中: 实心圈  $a \sim f$  代表临界层  $F_l$  中的个体, 空心圈代表前  $l-1$  层的个体,  $Z_1$  和  $Z_2$  为参考线, 种群的所有个体均位于  $R_1$  和  $R_2$  区域内. 假设需要从  $F_l$  中选择2个个体或淘汰4个个体, 即  $K = 2, Y = 4$ . 首先进行选择操作. 从图1中可看出  $Z_1$  在  $l-1$  层与5个个体相关联, 而  $Z_2$  则与6个个体相关联, 因此选择具有最小小生境数的区域  $R_1$ . 其中  $a, b$  和  $c$  都位于  $R_1$  中, 分别计算它们的AAD, 个体  $b$  具有最小距离值, 因此被选中进入下一代种群, 并且区域  $R_1$  的小生境数加1, 同时在下一选择操作中将不考虑个体  $b$ . 然后进行淘汰操作. 经过选择操作后, 两个区域有着相同的小生境数6, 所以随机选择一个区域. 假设选择  $R_2$  区域, 这里含有3个个体:  $d, e$  与  $f$ , 淘汰具有最大自适应聚合距离的个体  $f$ , 并且后续操作将不再被考虑. 接下来再次进行选择操作, 由于两个区域有着相同的小生境数, 随机选择一个区域. 若选择区域  $R_2$ , 则具有最小距离值的个体  $e$  被选入下一代种群. 最后, 选择所需的2个个体进入下一代.

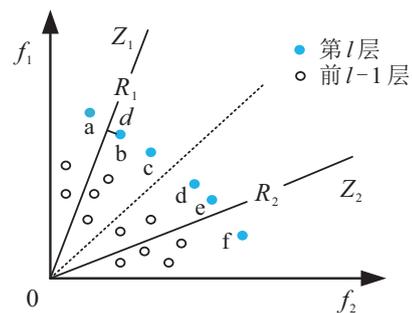


图1 基于自适应聚合距离的精英选择示意

## 2.4 算法复杂度分析

本文算法的复杂度主要表现在环境选择过程中. 对规模为 $2N$ 、目标维度为 $M$ 的种群,在最坏的情况下,即每个解互不支配,参考点非支配排序(算法2第1行)计算复杂度为 $O(MN^2)$ ;归一化操作(算法2第11行)的计算复杂度为 $O(MN)$ ;关联操作(算法2第14行)的计算复杂度为 $O(MNH)$ ;算法3中改进的小生境保留策略最坏情况下的计算复杂度为 $O(MN^2)$ . 综合以上分析,算法MOEA-AAD运行一代需要的最大计算复杂度是 $O(MN^2)$ .

## 2.5 讨论

近些年,人们提出了不少基于NSGA-III框架的改进算法,此处分析并讨论MOEA-AAD与AR-NSGA-III<sup>[22]</sup>、 $\theta$ -DEA的异同. 3种算法的最大相似点为产生一组均匀分布的参考点指导进化,并且都采取非支配排序对种群进行等级划分.

1) MOEA-AAD与AR-NSGA-III的比较:二者同样采用小生境保护策略,但它们的选择机制不同. 前者拥有选择与淘汰操作,而后者仅有选择操作. 对于 $F_1$ 中个体的选择策略,MOEA-AAD采用AAD距离进行评估,而AR-NSGA-III采用垂直距离.

2) MOEA-AAD与 $\theta$ -DEA的比较:两种算法同样引入惩罚距离作为对个体的评价机制. 不同之处是:MOEA-AAD的惩罚参数能够随着进化过程动态调整,而 $\theta$ -DEA则是固定的. 对于种群分层的方式,前者利用参考点支配,而后者则利用 $\theta$ 支配进行.

## 3 实验与分析

为验证所提出算法的有效性,实验中选择DTLZ<sup>[23]</sup>、WFG<sup>[24]</sup>和MaF<sup>[25]</sup>三个著名的多目标进化算法测试函数集进行仿真实验,将所提出算法与当前性能优异的多目标进化算法AR-NSGA-III<sup>[22]</sup>、MultiGPO<sup>[26]</sup>、CARV-MOEA<sup>[27]</sup>和 $\theta$ -DEA<sup>[21]</sup>进行对比,并在多目标进化软件包PlatEMO<sup>[28]</sup>上进行实验.

### 3.1 实验设置

1) 种群规模. NSGA-III、AR-NSGA-III等算法的种群规模与均分分布的权重向量有关. 目标个数为 $\{3, 5, 8, 10, 12\}$ ,分别对应的种群规模为 $\{91, 210, 156, 275, 364\}$ .

2) 进化算子参数设置. 交叉概率 $p_c = 1$ ,变异概率 $p_m = 1/D$ , $D$ 为决策空间的维度,交叉和变异的分布参数分别为 $\eta_c = 20$ , $\eta_m = 30$ .

3) 显著性检验. 每个算法在测试问题上均独立运行20次,采用显著性水平为5%的Wilcoxon秩和检验对实验进行比较.

4) 性能指标. 采用反世代距离IGD<sup>[29]</sup>和超体积HV<sup>[30]</sup>来验证算法的收敛性和分布性.

5) 对比算法参数设置. MultiGPO的扩展角度 $\varphi = 30$ ;CARV-MOEA惩罚因子变化率 $\alpha = 2$ ,参考向量自适应频率 $f = 0.1$ ; $\theta$ -DEA惩罚参数 $\theta = 5$ .

6) 算法终止条件. 对于DTLZ和WFG系列问题,函数评价次数(FE)设为40 000;对于MaF系列问题,5、8目标问题设为80 000,而10和12目标问题则设为140 000和200 000,运行代数 $T = FE/N$ .

### 3.2 实验结果分析

表1为DTLZ和WFG系列测试函数的IGD平均值和标准差统计结果,表现最佳的结果用粗体字表示. 表1中的“+”“-”“=”分别代表经过秩和检验后显著优于、显著劣于、无明显优劣于MOEA-AAD. 从表1中可以看出,本文算法在84个测试实例中获得34个最好结果,而AR-NSGA-III、MultiGPO、CARV-MOEA和 $\theta$ -DEA的最好结果分别有4、23、11和12个,从而表明MOEA-AAD算法具有很好的收敛性和分布性,具备良好的竞争力.

#### 3.2.1 DTLZ问题上的实验结果分析

DTLZ2-3具有规则的PF,MOEA-AAD在处理该类前沿时效果较好,整体上表现优异. MOEA-AAD在DTLZ2上的表现仅次于 $\theta$ -DEA,在DTLZ3上则略逊于MultiGPO. DTLZ5-6有着退化的不规则PF,MOEA-AAD能较好地处理这类问题. 对于DTLZ5,MOEA-AAD在8维和10维测试问题上取得了最优结果,MultiGPO则在剩余的目标维数上表现优异;对于DTLZ6,MOEA-AAD获得了最优的整体性能. DTLZ7的PF带有混合和不连续结构,尽管其他对比算法在某一特定维数上表现优异,但综合所有实验结果,MOEA-AAD的整体性能最好.

图2为MOEA-AAD在DTLZ3上的IGD变化曲线及其惩罚参数变化曲线. 在图2(a)中的进化前期(大约评价次数 $<15 000$ ),IGD值下降很快,说明分散的种群个体快速收敛,此时算法强调收敛性;进化后期(大约评价次数 $>15 000$ ),IGD值收敛速度减缓. 从图2(b)中可以看出惩罚参数 $\theta$ 在此阶段明显增大,此时算法强调个体的分布性. 由此可知,在DTLZ3的进化过程中,MOEA-AAD算法在不同阶段的侧重点不同,在前15 000次评价次数中侧重于个体的收敛性,而在15 000次评价次数后则侧重于个体的分布性.

#### 3.2.2 WFG问题上的实验结果分析

WFG1-3有着不规则的PF的测试问题,WFG1的PF带有混合与有偏结构,MOEA-AAD的整体性能优

表1 5种算法在DTLZ和WFG测试问题上获得的IGD平均值与标准差

问题	M	AR-NSGA-III	MultiGPO	CARV-MOEA	$\theta$ -DEA	MOEA-AAD
DTLZ2	3	6.429 4e-2 (1.61e-3) -	6.600 5e-2 (2.06e-3) -	5.5794e-2 (3.42e-6) +	<b>5.446 6e-2 (2.25e-6) +</b>	5.650 5e-2 (5.40e-4)
	5	1.739 9e-1 (1.05e-3) -	1.760 2e-1 (1.47e-3) -	1.7631e-1 (5.34e-4) -	<b>1.651 3e-1 (8.28e-5) =</b>	1.667 5e-1 (6.73e-4)
	8	4.114 1e-1 (3.81e-2) -	3.582 6e-1 (2.27e-3) -	3.574 8e-1 (4.53e-3) -	<b>3.178 5e-1 (5.23e-4) +</b>	3.444 0e-1 (5.47e-3)
	10	5.082 7e-1 (6.13e-2) -	<b>4.109 8e-1 (1.65e-3) +</b>	4.481 0e-1 (3.64e-3) -	4.321 0e-1 (2.54e-3) =	4.343 6e-1 (3.14e-3)
DTLZ3	3	7.255 8e-2 (5.87e-3) +	<b>7.081 5e-2 (2.72e-3) +</b>	7.752 7e-2 (3.27e-1) +	9.752 7e-2 (1.72e-1) -	8.442 6e-2 (1.76e-1)
	5	2.756 4e+0 (2.17e+0) -	<b>2.996 5e-1 (2.65e-1) +</b>	3.473 1e-1 (4.16e-1) =	9.415 1e-1 (8.66e-1) -	3.492 7e-1 (3.63e-1)
	8	8.141 3e+0 (2.56e+0) -	<b>9.656 7e-1 (8.32e-1) +</b>	1.098 9e+0 (8.64e-1) =	1.313 2e+0 (9.15e-1) -	1.104 8e+0 (9.16e-1)
	10	2.170 3e+1 (8.96e+0) -	<b>1.675 5e+0 (1.30e+0) +</b>	2.451 3e+0 (1.89e+0) -	3.256 3e+0 (1.51e+0) -	1.902 1e+0 (1.36e+0)
DTLZ5	3	1.192 9e-2 (1.78e-3) +	<b>5.909 9e-3 (4.80e-4) +</b>	4.891 3e-2 (2.28e-3) -	3.420 0e-2 (2.28e-3) -	3.2756e-2 (1.69e-3)
	5	9.297 0e-2 (2.22e-2) =	<b>3.662 2e-2 (9.77e-3) +</b>	1.234 7e-1 (5.31e-2) -	2.190 8e-1 (6.91e-2) -	9.308 9e-2 (2.25e-2)
	8	2.575 3e-1 (5.29e-2) -	1.821 6e-1 (3.26e-2) -	1.849 1e-1 (5.43e-2) -	1.700 7e-1 (5.82e-2) -	<b>1.549 4e-1 (2.56e-2)</b>
	10	3.075 9e-1 (3.51e-2) -	2.104 2e-1 (3.12e-2) -	2.0316e-1 (4.89e-2) -	1.639 2e-1 (4.14e-2) -	<b>1.602 8e-1 (2.26e-2)</b>
DTLZ6	3	1.735 8e-2 (1.60e-3) -	5.745 6e-2 (3.99e-3) -	3.154 7e-2 (5.34e-3) -	4.008 8e-2 (6.25e-3) -	<b>1.662 7e-2 (5.95e-2)</b>
	5	9.874 2e-1 (8.50e-2) -	<b>5.786 9e-2 (1.26e-2) +</b>	1.884 3e-1 (4.31e-2) +	2.859 9e-1 (6.15e-2) -	2.315 4e-1 (1.85e-1)
	8	2.488 4e-1 (3.06e-2) -	2.459 7e-1 (7.82e-2) -	2.247 9e-1 (4.33e-2) -	2.238 7e-1 (5.66e-2) -	<b>1.995 9e-1 (1.33e-1)</b>
	10	2.051 1e+0 (6.84e-1) -	3.521 0e-1 (3.07e-1) -	3.094 5e-1 (6.17e-2) -	2.591 7e-1 (6.17e-2) -	<b>1.461 9e-1 (3.63e-2)</b>
DTLZ7	3	<b>9.041 9e-2 (6.79e-2) +</b>	1.562 3e-1 (1.19e-1) -	1.354 3e-1 (5.86e-2) -	1.086 6e-1 (5.99e-2) -	9.766 3e-2 (6.15e-3)
	5	2.959 4e-1 (8.44e-3) -	2.986 6e-1 (3.74e-2) -	2.954 6e-1 (2.31e-2) -	3.065 5e-1 (2.01e-2) -	<b>2.875 6e-1 (1.02e-2)</b>
	8	1.120 9e+0 (5.25e-2) -	<b>6.100 7e-1 (7.50e-3) +</b>	6.464 1e-1 (5.82e-2) -	7.469 0e-1 (5.13e-2) -	1.088 5e+0 (9.18e-2)
	10	2.207 1e+0 (2.31e-1) -	<b>8.854 6e-1 (1.92e-2) +</b>	1.131 5e+0 (3.63e-1) +	9.122 1e-1 (1.00e-1) +	2.188 0e+0 (6.88e-1)
WFG1	3	2.243 9e-1 (2.80e-2) +	2.493 1e-1 (2.17e-2) +	<b>1.746 1e-1 (2.74e-2) +</b>	1.77 20e-1 (2.53e-2) +	2.667 7e-1 (9.32e-2)
	5	6.738 6e-1 (6.74e-2) -	1.030 2e+0 (2.25e-1) -	5.236 4e-1 (4.31e-2) -	4.798 5e-1 (4.83e-2) -	<b>4.276 1e-1 (2.94e-2)</b>
	8	1.103 9e+0 (9.46e-2) -	1.466 5e+0 (1.48e-1) -	9.831 6e-1 (5.94e-2) -	9.386 1e-1 (4.40e-2) -	<b>9.053 6e-1 (6.92e-2)</b>
	10	1.768 2e+0 (9.05e-2) -	1.678 0e+0 (1.31e-1) -	1.654 6e+0 (6.20e-2) -	1.535 8e+0 (6.24e-2) -	<b>1.403 4e+0 (4.82e-2)</b>
WFG2	3	1.759 0e-1 (8.33e-3) -	3.087 2e-1 (1.46e-2) -	2.548 1e-1 (2.81e-2) -	<b>1.582 6e-1 (1.60e-3) =</b>	1.591 1e-1 (2.53e-3)
	5	3.993 7e-1 (8.81e-2) -	9.483 5e-1 (1.57e-1) -	3.876 1e-1 (4.38e-3) =	3.970 1e-1 (3.95e-3) -	<b>3.856 8e-1 (4.05e-3)</b>
	8	1.192 0e+0 (2.55e-1) -	1.666 5e+0 (1.63e-1) -	1.058 1e+0 (1.93e-1) -	1.056 0e+0 (1.71e-1) =	<b>1.007 6e+0 (2.50e-2)</b>
	10	1.211 0e+0 (1.10e-1) -	1.733 0e+0 (1.06e-1) -	1.207 6e+0 (1.77e-1) -	1.199 6e+0 (1.55e-1) -	<b>1.086 9e+0 (1.45e-2)</b>
WFG3	3	1.711 0e-1 (2.73e-2) -	<b>4.790 0e-2 (5.06e-3) +</b>	2.218 2e-1 (3.58e-2) -	1.383 1e-1 (2.39e-2) +	1.550 3e-1 (1.66e-2)
	5	4.348 1e-1 (4.33e-2) -	7.499 7e-1 (1.14e-1) -	5.812 8e-1 (3.50e-2) -	4.805 9e-1 (2.70e-2) -	<b>3.963 4e-1 (3.42e-2)</b>
	8	1.261 9e+0 (4.07e-1) =	2.593 7e+0 (7.37e-1) -	2.896 1e+0 (6.74e-1) -	1.296 7e+0 (1.71e-1) =	<b>1.258 5e+0 (8.74e-2)</b>
	10	1.624 8e+0 (2.35e-1) -	2.048 3e+0 (2.57e-1) -	1.851 1e+0 (1.49e-1) -	1.566 8e+0 (1.06e-1) =	<b>1.568 3e+0 (2.78e-2)</b>
WFG4	3	2.532 5e-1 (1.31e-2) -	3.307 5e-1 (1.92e-2) -	<b>2.194 1e-1 (2.91e-3) +</b>	2.309 3e-1 (3.10e-3) =	2.298 0e-1 (3.96e-3)
	5	9.793 0e-1 (1.75e-2) =	1.194 6e+0 (5.54e-2) -	1.349 4e+0 (8.45e-2) -	<b>9.647 9e-1 (1.42e-3) +</b>	9.755 3e-1 (4.59e-3)
	8	3.420 5e+0 (3.73e-1) -	3.362 3e+0 (1.42e-1) -	3.264 6e+0 (5.38e-2) -	<b>2.993 6e+0 (7.61e-3) =</b>	3.011 5e+0 (2.84e-2)
	10	4.678 4e+0 (5.47e-1) -	4.322 0e+0 (1.37e-1) +	<b>4.291 4e+0 (3.08e-2) +</b>	4.515 8e+0 (2.53e-2) =	4.518 2e+0 (2.79e-2)
WFG5	3	2.452 3e-1 (5.16e-3) -	3.192 8e-1 (1.99e-2) -	3.051 3e-1 (5.37e-5) -	2.598 9e-1 (7.04e-5) -	<b>2.372 6e-1 (4.05e-3)</b>
	5	<b>9.550 3e-1 (8.81e-3) +</b>	1.168 7e+0 (4.38e-2) -	9.661 2e-1 (4.34e-3) =	9.791 5e-1 (2.72e-3) -	9.659 6e-1 (4.54e-3)
	8	3.209 5e+0 (9.05e-2) -	3.147 8e+0 (5.34e-2) -	3.384 1e+0 (4.61e-2) -	<b>2.942 3e+0 (7.45e-3) =</b>	2.975 1e+0 (1.53e-2)
	10	4.619 8e+0 (4.06e-1) -	<b>4.239 5e+0 (7.60e-2) +</b>	4.443 7e+0 (4.33e-2) =	4.439 0e+0 (2.80e-2) =	4.485 7e+0 (2.13e-2)
WFG6	3	2.634 5e-1 (2.21e-2) -	3.504 3e-1 (1.55e-2) -	3.435 8e-1 (2.33e-2) -	<b>2.390 7e-1 (1.07e-2) +</b>	2.488 1e-1 (9.50e-3)
	5	1.012 3e+0 (2.75e-2) -	1.343 2e+0 (4.14e-2) -	1.584 1e+0 (5.65e-2) -	9.880 2e-1 (6.94e-3) -	<b>9.793 8e-1 (6.34e-3)</b>
	8	3.978 8e+0 (7.33e-1) -	3.622 4e+0 (1.89e-1) -	<b>2.268 4e+0 (2.61e-2) +</b>	2.969 4e+0 (9.31e-3) =	2.994 7e+0 (1.93e-2)
	10	4.854 9e+0 (5.05e-2) -	4.778 2e+0 (1.63e-2) -	4.615 1e+0 (2.94e-2) =	4.590 1e+0 (1.74e-2) =	<b>4.589 6e+0 (2.18e-2)</b>
+ / - / =		5/36/3	14/30/0	8/30/6	7/24/13	

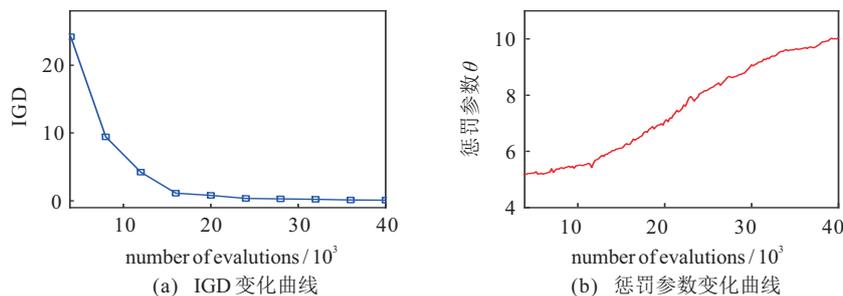


图2 MOEA-AAD在DTLZ3上IGD值变化曲线及其惩罚参数变化曲线

异,只在3目标上略逊于其他对比算法. WFG2为多模态、不连续且不可分解的PF,由实验结果可知,MOEA-AAD处理该问题的能力最好. WFG3的PF较为复杂,有着连续和退化特性,MOEA-AAD的IGD值除在3目标上略逊于 $\theta$ -DEA外,其他均优于对比算法. WFG4-6被设定为具有凹形面的PF,MOEA-AAD在WFG4-5上的性能表现略差于CARV-MOEA和 $\theta$ -DEA,在WFG6的5目标和10目标上取得了最好的IGD指标. 综合统计结果,MOEA-AAD在WFG测试集上整体性能效果优越.

### 3.2.3 MaF问题上的实验结果分析

MaF1有着线性的PF,算法在此类问题上表现不佳,仅优于AR-NSGA-III. MaF2具有凹形面的PF,MOEA-AAD在8、10维目标上的表现尤为突出. MaF3具有凸形和多模态特性的PF,在此问题上MultiGPO的表现最优. MaF4有着凹形和多模态的PF,CARV-MOEA的结果最好. MaF5的PF带有凸形和有偏结构,MOEA-AAD的性能优于对比算法. MaF6具有凹形且退化的PF,MOEA-AAD在10、12目标上取得了全局最优. MaF7有着混合、多模态的不规则PF,MOEA-AAD的性能较差. MaF9的PF具有线性和退化的特点,本文算法逊色于MultiGPO. MaF10和MaF11是比较难处理的问题,MOEA-AAD在IGD指标上取得了最优的结果. 由实验可知,MOEA-AAD在MaF测试集上展现出优异的性能.

综合实验结果及上述分析可得,MOEA-AAD算法在以上测试集上具有较为明显的统计意义上的性能优势,并且在处理高维多目标优化时有着较强的竞争力,具备优良的处理复杂不规则Pareto前沿的能力,证明了其能够很好地平衡收敛性与分布性.

为了对实验结果有一个更加直观地反映,图3中(a)、(b)为5种算法在WF3和MaF11上解集的IGD值折线图. 从图3中可看出:MOEA-AAD在WFG3上,MOEA-AAD除在3目标上表现不如MultiGPO和 $\theta$ -DEA外,其余目标均优于对比算法;而对于MaF11,MOEA-AAD取得了最优的整体性能,具备很强的竞争力. 图3中(c)、(d)为各算法在DTLZ5和MaF10上IGD值的变化曲线. 可以看出,在迭代前期阶段,MOEA-AAD与 $\theta$ -DEA两种算法的收敛曲线下降明显,这说明引入基于惩罚的聚合函数作为评价机制可以增强种群的搜索能力. 在迭代后期阶段,IGD值稳定在一个较小的值上,说明改进的小生境选择策略增强了种群的分布性,从而表明了改进算法的有效性. AR-NSGA-III的实验结果较差,收敛速度慢且精度不准,这是因为它在高维度问题上缺乏合理的选择机制,使种群靠近前沿面. MultiGPO和CARV-MOEA获得的结果不佳,出现收敛速度缓慢且收敛方向不稳定的情况.  $\theta$ -DEA缺少自适应调节策略,导致进化后期种群个体的分布性较低. 而MOEA-AAD的性能则优于其他对比算法.

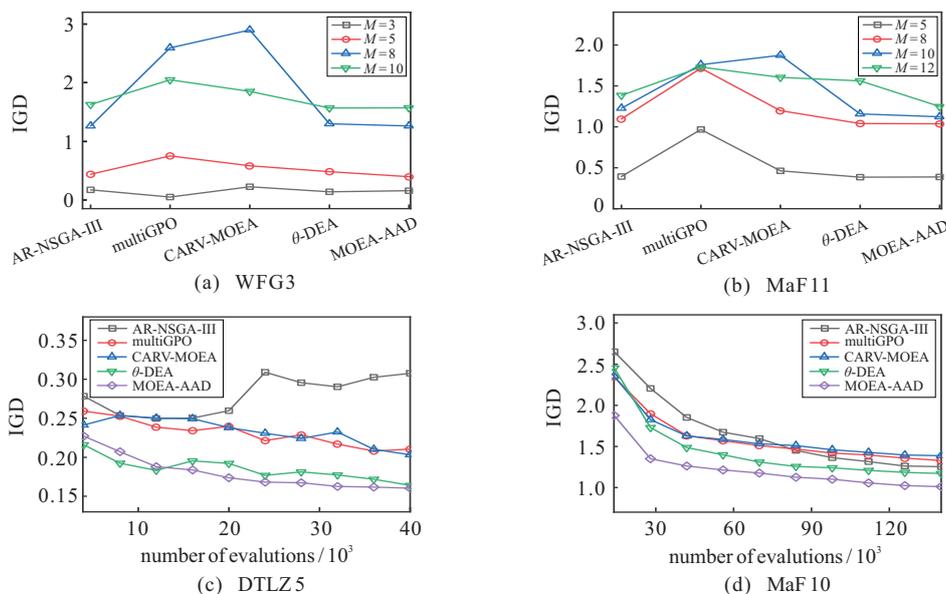


图3 5种算法在4个测试问题各目标上获得解集的IGD值和10目标上的IGD值变化曲线

为了更加直观地对比算法的最终解集在高维空间上的分布情况,图4绘制了4种算法在5目标DTLZ7以及10目标WFG1上IGD值最接近平均值

的一组数据的平行坐标图,由于AR-NSGA-III的性能不佳,此处不再给出最终解集图. 由图4可以看出:对于DTLZ7,MultiGPO和CARV-MOEA获得了较优

的最终解集;  $\theta$ -DEA和MEOA-AAD均取得了满意的结果. 对于WFG1, CARV-MOEA、 $\theta$ -DEA和MEOA-

AAD均获得了较优的解集; 而MultiGPO所得的解集的收敛性较差, 只收敛到前沿面的某部分区域上.

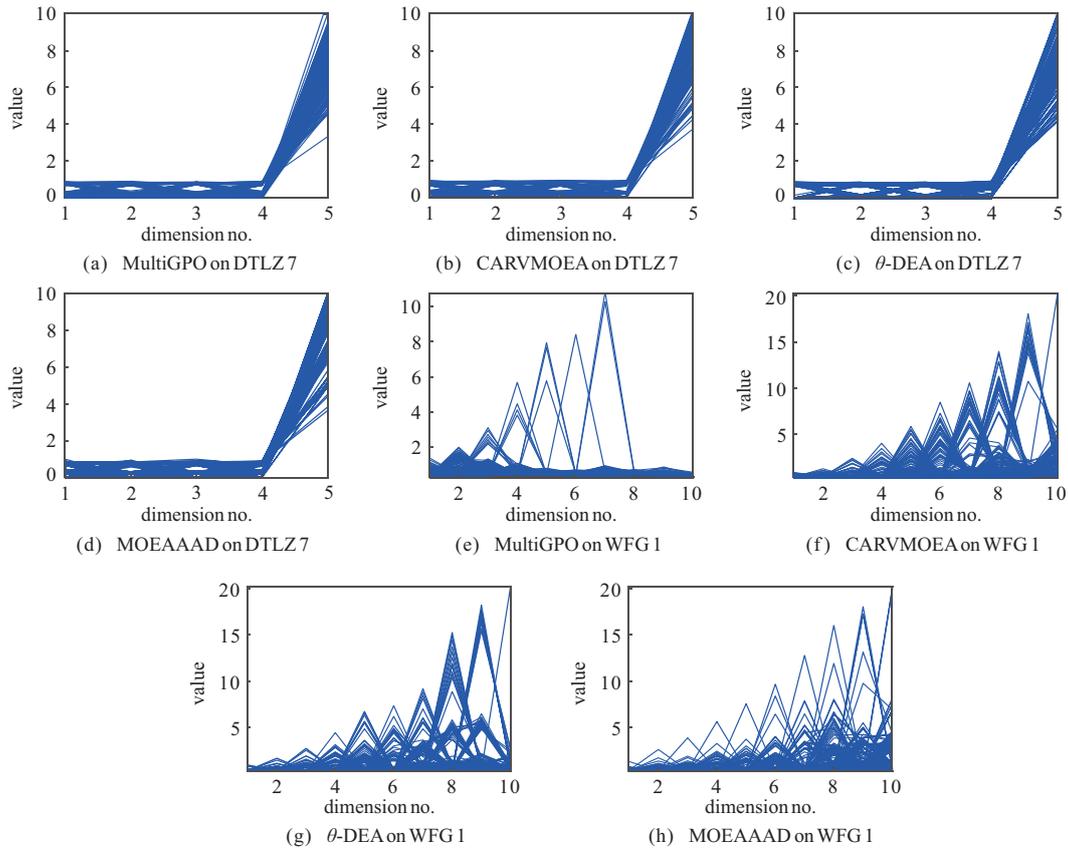


图4 4种算法在5目标DTLZ7和10目标WFG1问题上的最终解集

### 3.3 参数影响分析

为分析自定义惩罚因子 $\delta$ 对算法性能的影响, 本文选取6个不同的值, 即1、3、5、8、12、15并对WFG6和MaF2进行实验. 每个测试函数独立运行20次, 记录IGD平均值, 如图5所示.

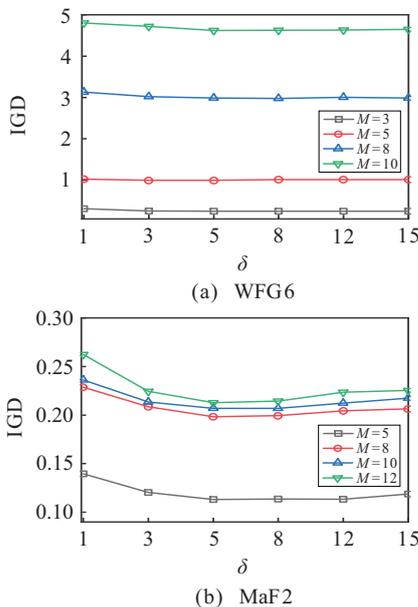


图5 MOEA-AAD在WFG6和MaF2上获得IGD平均值

由图5可观察到: 当 $\delta = 1$ 时效果不佳, 其原因在于 $\delta$ 值过小会导致进化初期出现收敛过度, 到进化末期分布性不理想; 当 $\delta = 3$ 时, 算法的性能明显转优; 而当 $\delta = 5$ 时, 算法获得最优的结果; 随着 $\delta$ 值增大, 实验结果基本不发生较大的波动. 同时可以发现,  $\delta$ 在3个测试函数上的变化趋势大体一致, 由此说明 $\delta$ 对不同问题特性的敏感度并不高. 因此,  $\delta$ 取 $[3, 8]$ 之间的值能获得相对较好的结果. 本文惩罚因子设为5.

## 4 实际应用

本文将MOEA-AAD算法应用到多目标旅行商问题<sup>[31]</sup>和水资源管理问题<sup>[32]</sup>上. 由于实际问题难以获取真实PF, 这里采用HV来衡量各算法的效果.

### 4.1 多目标旅行商问题

多目标旅行商问题 (multi-objective travelling salesman problem, MOTSP) 是经典旅行商问题的扩展, 其数学表达式及具体参数设置可参考文献<sup>[31]</sup>.

该问题目标数量分别为3、5、8、10以及12, 对应的种群规模设为120、105、128、110以及90, 决策变量设为30, 最大评价次数设定为50 000次, 各算法独立运行20次.

仿真结果表明: MOEA-AAD 在 10 和 12 目标上表现最好; MultiGPO 算法在 3 和 5 目标问题上获得了最优结果; 而对于 8 目标问题,  $\theta$ -DEA 是性能最好的算法. 总之, 对于 5 个不同的 MOTSP 测试目标, MultiGPO 和 MOEA-AAD 算法表现较好.

#### 4.2 水资源管理问题

水资源管理问题 (water resources planning problem, WRPP) 是多目标领域中较为重要的问题, 详细数学描述见文献 [32].

该问题种群规模设为 105, 最大评价次数设定为 50 000 次, 各算法独立运行 20 次.

仿真结果表明, MOEA-AAD 的表现最优异, 而 MultiGPO 和 CARV-MOEA 分别次之. 整体来看, 对于水资源管理问题, MOEA-AAD 获得了最优结果, 比其他算法更具竞争力.

### 5 结论

本文提出了一种基于自适应聚合距离的多目标进化算法, 用于更好地实现种群的收敛性与分布性之间的平衡. 算法采用参考点支配方式, 籍此在高维空间中提高种群收敛到真实前沿面的能力; 设计一种自适应聚合距离, 在不同的进化过程中自适应调节收敛性和分布性比重, 并在小生境保留策略中设计一种带有选择与淘汰机制方案, 有效地选出优异的个体进入下一代. 在经典测试函数集 DTLZ、WFG 和 MaF 的部分测试函数上的实验仿真结果表明: 相比于现阶段性能优越的算法, MOEA-AAD 在大多数情况下表现优异, 能较好地处理复杂 Pareto 前沿问题, 获得的解集具备良好的收敛性和分布性. 在两个实际应用中, 本文算法同样具有更好的性能, 进一步表明了 MOEA-AAD 在解决多目标问题上的竞争力. 在后续的研究工作中, 将会把本文所提出算法的应用扩展至大规模优化问题中.

#### 参考文献 (References)

- [1] Sassi J, Alaya I, Borne P, et al. A decomposition-based artificial bee colony algorithm for the multi-objective flexible jobshop scheduling problem[J]. *Engineering Optimization*, 2022, 54(3): 524-538.
- [2] 张国富, 陆淑君, 苏兆品, 等. 化工园区应急物资多目标分配问题建模与求解[J]. *控制与决策*, 2022, 37(4): 962-972.  
(Zhang G F, Lu S J, Su Z P, et al. Modeling and solving multi-objective emergency resource allocation in chemical industrial parks[J]. *Control and Decision*, 2022, 37(4): 962-972.)
- [3] 王丽萍, 任宇, 邱启仓, 等. 多目标进化算法性能评价指标研究综述[J]. *计算机学报*, 2021, 44(8): 1590-1619.  
(Wang L P, Ren Y, Qiu Q C, et al. Survey on performance indicators for multi-objective evolutionary algorithms[J]. *Chinese Journal of Computers*, 2021, 44(8): 1590-1619.)
- [4] Deb K, Pratap A, Agarwal S, et al. A fast and elitist multiobjective genetic algorithm: NSGA-II[J]. *IEEE Transactions on Evolutionary Computation*, 2002, 6(2): 182-197.
- [5] 封文清, 巩敦卫. 基于在线感知 Pareto 前沿划分目标空间的多目标进化优化[J]. *自动化学报*, 2020, 46(8): 1628-1643.  
(Feng W Q, Gong D W. Multi-objective evolutionary optimization with objective space partition based on online perception of Pareto front[J]. *Acta Automatica Sinica*, 2020, 46(8): 1628-1643.)
- [6] Yang W S, Chen L, Wang Y, et al. A reference points and intuitionistic fuzzy dominance based particle swarm algorithm for multi/many-objective optimization[J]. *Applied Intelligence*, 2020, 50(4): 1133-1154.
- [7] Li L, Li G P, Chang L. A many-objective particle swarm optimization with grid dominance ranking and clustering[J]. *Applied Soft Computing*, 2020, 96: 106661.
- [8] Liu Y, Zhu N B, Li K L, et al. An angle dominance criterion for evolutionary many-objective optimization[J]. *Information Sciences*, 2020, 509: 376-399.
- [9] Zhang Q F, Li H. MOEA/D: A multiobjective evolutionary algorithm based on decomposition[J]. *IEEE Transactions on Evolutionary Computation*, 2007, 11(6): 712-731.
- [10] Deb K, Jain H. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach—Part I: Solving problems with box constraints[J]. *IEEE Transactions on Evolutionary Computation*, 2014, 18(4): 577-601.
- [11] Sun Y H, Xiao K L, Wang S Q, et al. An evolutionary many-objective algorithm based on decomposition and hierarchical clustering selection[J]. *Applied Intelligence*, 2022, 52(8): 8464-8509.
- [12] Chen J X, Ding J L, Tan K C, et al. A decomposition-based evolutionary algorithm for scalable multi/many-objective optimization[J]. *Memetic Computing*, 2021, 13(3): 413-432.
- [13] Yang F, Xu L, Chu X K, et al. A new dominance relation based on convergence indicators and niching for many-objective optimization[J]. *Applied Intelligence*, 2021, 51(8): 5525-5542.
- [14] Liang Z P, Luo T T, Hu K F, et al. An indicator-based many-objective evolutionary algorithm with boundary protection[J]. *IEEE Transactions on Cybernetics*, 2021, 51(9): 4553-4566.
- [15] Sun W J, Li J H. A strengthened diversity indicator and reference vector-based evolutionary algorithm for

- many-objective optimization[J]. *Soft Computing*, 2021, 25(15): 10257-10273.
- [16] Dhiman G, Soni M, Pandey H M, et al. A novel hybrid hypervolume indicator and reference vector adaptation strategies based evolutionary algorithm for many-objective optimization[J]. *Engineering with Computers*, 2021, 37(4): 3017-3035.
- [17] Shen J T, Wang P, Wang X J. A controlled strengthened dominance relation for evolutionary many-objective optimization[J]. *IEEE Transactions on Cybernetics*, 2022, 52(5): 3645-3657.
- [18] Gu Q H, Chen H Y, Chen L, et al. A many-objective evolutionary algorithm with reference points-based strengthened dominance relation[J]. *Information Sciences*, 2021, 554: 236-255.
- [19] Elarbi M, Bechikh S, Gupta A, et al. A new decomposition-based NSGA-II for many-objective optimization[J]. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2018, 48(7): 1191-1210.
- [20] Chen L, Deb K, Liu H L, et al. Effect of objective normalization and penalty parameter on penalty boundary intersection decomposition-based evolutionary many-objective optimization algorithms[J]. *Evolutionary Computation*, 2021, 29(1): 157-186.
- [21] Yuan Y, Xu H, Wang B, et al. A new dominance relation-based evolutionary algorithm for many-objective optimization[J]. *IEEE Transactions on Evolutionary Computation*, 2015, 20(1): 16-37.
- [22] 耿焕同, 戴中斌, 王天雷, 等. 基于参考点选择策略的改进型NSGA-III算法[J]. *模式识别与人工智能*, 2020, 33(3): 191-201.  
(Geng H T, Dai Z B, Wang T L, et al. Improved NSGA-III algorithm based on reference point selection strategy[J]. *Pattern Recognition and Artificial Intelligence*, 2020, 33(3): 191-201.)
- [23] Deb K, Thiele L, Laumanns M, et al. Scalable test problems for evolutionary multiobjective optimization[C]. *Advanced Information and Knowledge Processing*. London: Springer-Verlag, 2005: 105-145.
- [24] Huband S, Hingston P, Barone L, et al. A review of multiobjective test problems and a scalable test problem toolkit[J]. *IEEE Transactions on Evolutionary Computation*, 2006, 10(5): 477-506.
- [25] Cheng R, Li M Q, Tian Y, et al. A benchmark test suite for evolutionary many-objective optimization[J]. *Complex & Intelligent Systems*, 2017, 3(1): 67-81.
- [26] Zhu S W, Xu L H, Goodman E D, et al. A new many-objective evolutionary algorithm based on generalized Pareto dominance[J]. *IEEE Transactions on Cybernetics*, 2022, 52(8): 7776-7790.
- [27] Liu Q Q, Jin Y C, Heiderich M, et al. Coordinated adaptation of reference vectors and scalarizing functions in evolutionary many-objective optimization[J]. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2023, 53(2): 763-775.
- [28] Tian Y, Cheng R, Zhang X Y, et al. PlatEMO: A Matlab platform for evolutionary multi-objective optimization[J]. *IEEE Computational Intelligence Magazine*, 2017, 12(4): 73-87.
- [29] Bosman P A N, Thierens D. The balance between proximity and diversity in multiobjective evolutionary algorithms[J]. *IEEE Transactions on Evolutionary Computation*, 2003, 7(2): 174-188.
- [30] Zitzler E, Thiele L. Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach[J]. *IEEE Transactions on Evolutionary Computation*, 1999, 3(4): 257-271.
- [31] Corne D W, Knowles J D. Techniques for highly multiobjective optimisation: Some nondominated points are better than others[C]. *Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation*. London, 2007: 773-780.
- [32] Xiang Y, Peng J, Zhou Y R, et al. An angle based constrained many-objective evolutionary algorithm[J]. *Applied Intelligence*, 2017, 47(3): 705-720.

### 作者简介

曾亮(1980—), 男, 教授, 博士, 从事机器视觉与人工智能、优化计算方法、调度优化等研究, E-mail: zengliang@hbut.edu.cn;

曾维钧(1997—), 男, 硕士生, 从事进化算法的研究, E-mail: 1095190375@qq.com;

李燕燕(1998—), 女, 硕士生, 从事优化调度的研究, E-mail: 1047880224@qq.com;

全睿(1984—), 男, 副教授, 博士, 从事新能源汽车故障诊断的研究, E-mail: quan\_rui@126.com;

王珊珊(1982—), 女, 副教授, 博士, 从事图像处理的研究, E-mail: wangshanshan@hbut.edu.cn.