

# 基于角度惩罚距离精英选择策略的 偏好高维目标优化算法

王丽萍<sup>1)</sup> 章鸣雷<sup>1)</sup> 邱飞岳<sup>2),3)</sup> 江 波<sup>2),3)</sup>

<sup>1)</sup>(浙江工业大学信息智能与决策优化研究所 杭州 310023)

<sup>2)</sup>(浙江工业大学教育科学与技术学院 杭州 310023)

<sup>3)</sup>(浙江工业大学现代教育技术研究所 杭州 310023)

**摘 要** 基于决策者偏好的高维目标优化算法能有效集中算法资源和减小搜索空间,是处理高维目标优化问题的有效途径之一. 现有研究发现,参考点位置选择对算法性能影响显著,位于极端位置的参考点容易引发算法不收敛;同时,算法多样性在种群逼近 Pareto 前沿的过程中反复遭到破坏. 为解决以上问题,该文提出一种基于角度惩罚距离精英选择策略的偏好高维目标优化算法. 该算法将决策者偏好信息融入到基于分解的多目标优化算法中,提出偏好向量生成策略,消除算法收敛性对参考点位置的敏感性;同时引入角度惩罚距离(APD)机制,分析该机制在算法搜索后期存在种群退化、收敛放缓等缺陷的基础上,提出 APD 精英选择策略,通过有效分配算法资源,平衡算法收敛性和多样性. 算法性能对比实验中,将该文提出的算法与  $g$ -占优、 $r$ -占优、双极偏好占优以及 MOEA/D-PRE 在 3 至 10 维 DTLZ1-4 测试问题上进行性能测试. 实验结果表明,该文提出的偏好算法所求解集能够有效反映决策者的偏好信息,并且在高维目标优化问题上,所提算法在偏好区域求得解集的收敛性和均匀性更优.

**关键词** 高维目标优化;偏好向量;角度惩罚距离;精英选择;进化算法  
**中图法分类号** TP391 **DOI 号** 10.11897/SP.J.1016.2018.00236

## Many-Objective Optimization Algorithm with Preference Based on the Angle Penalty Distance Elite Selection Strategy

WANG Li-Ping<sup>1)</sup> ZHANG Ming-Lei<sup>1)</sup> QIU Fei-Yue<sup>2),3)</sup> JIANG Bo<sup>2),3)</sup>

<sup>1)</sup>(Institute of Information Intelligence and Decision Optimization, Zhejiang University of Technology, Hangzhou 310023)

<sup>2)</sup>(College of Educational Science and Technology, Zhejiang University of Technology, Hangzhou 310023)

<sup>3)</sup>(Institute of Modern Educational Technology, Zhejiang University of Technology, Hangzhou 310023)

**Abstract** Many-objective optimization problem is the hotspot and difficulty in the field of multi-objective optimization. With the increase of the target dimension, the weakness of traditional multi-objective evolutionary algorithm appears gradually; the proportion of non-dominated solutions in the population increases rapidly, the population size required to cover the entire Pareto frontier grows exponentially, and the balance of convergence and diversity for the algorithm is hard to maintain. The many-objective optimization algorithm based on decision maker's preference is promising in solving many-objective optimization problems (MaOP) because of its ability in concentrating computational resources and reducing search space during searching process. Existing work found that the performance of this kind of algorithm is sensitive to the location of reference

收稿日期:2016-06-21;在线出版日期:2017-06-23. 本课题得到国家自然科学基金(61472366,61379077,61503340)、浙江省自然科学基金(LY17F020022,LQ16F030008)资助. 王丽萍,女,1964年生,教授,博士生导师,主要研究领域为信息处理、决策优化. E-mail: wlp@zjut.edu.cn. 章鸣雷,男,1992年生,硕士研究生,主要研究方向为智能进化算法、决策优化. 邱飞岳(通信作者),男,1965年生,博士,教授,主要研究领域为智能控制、优化理论与方法. E-mail: qfy@zjut.edu.cn. 江 波,男,1985年生,博士,副教授,主要研究方向为智能优化与控制.

points. Particularly, it is frequently observed that the non-convergence happened when the reference points are in the extreme region of Pareto front. Besides, the diversity of population is repeatedly destroyed when approximating the Pareto front. To solve these problems, a many-objective optimization algorithm with preference based on the angle penalty distance elite selection strategy is proposed in this work. Firstly, the proposed algorithm incorporates the preference information into the decomposition based multi-objective optimization algorithm, producing a preference vector generation strategy. To reduce influence of the reference point position to algorithm convergence, in the proposed preference vector generation method, the weight vectors are scaled based on the radius parameter of preference region  $\epsilon$  given by the decision maker to produce preference vectors. Secondly, an angle punishment distance (APD) based elite selection strategy is introduced to allocate computational resource more adaptively which aiming at balancing the convergence and diversity, after analyzing the defects of population degradation and slow convergence in the later stage of algorithm search. In the proposed strategy, the algorithm focuses on the convergence of the population in the early stage of searching, taking the distance of individuals from the origin of the target space as the screening criterion, which forces the population to approach the Pareto front rapidly, and greatly saves the computational cost of the algorithm in the early search of the many-objective target space. In the latter stage of the search, the algorithm focuses on the diversity of the population, slowing down the convergence rate of the population toward the Pareto front, and eliminating the Pareto dominated solutions and punishing the individuals who deviates from the direction of the preference vector, to maintain the high diversity of the population in the many-objective target space and the high uniformity of the solution set. The simulation results show that the APD elite selection strategy removes dominated individuals in the sub population, which makes the algorithm have more computational resources for maintaining population diversity and uniformity of set distribution. In the performance comparison experiment, the proposed algorithm was compared to four state-of-the-art algorithms, i. e., g-dominated, R-dominance, bipolar preferences dominance and MOEA/D-PRE algorithm, on DTLZ series benchmarks with 3–10 objectives. The experimental results show that the proposed algorithm not only reflect the preference information of decision maker effectively, but also outperforms the compared algorithms on the algorithm convergence and solution distribution.

**Keywords** many-objective optimization; preference vector; angle penalty distance; elite selection; evolutionary algorithm

## 1 引言

高维目标优化是目前多目标优化领域的研究热点和难点<sup>[1]</sup>,且广泛应用于物流工程<sup>[2-3]</sup>、机械工程<sup>[4]</sup>、网络节点<sup>[5]</sup>和自动化控制<sup>[6]</sup>等实际问题中,具有重要的研究价值.传统的多目标优化算法是基于 Pareto 支配关系的,代表性算法有 Deb 等人<sup>[7]</sup>提出的快速非支配排序算法(Fast and Elitist Multi-objective Genetic Algorithm, NSGA-II)和 Zitzler

等人<sup>[8]</sup>提出的强化 Pareto 进化算法(Improving the Performance of the Strength Pareto Evolutionary Algorithm, SPEA2),其主要思想是在目标空间中搜索得到逼近整个 Pareto 前沿的非支配解集,从而获得目标问题的近似最优解. Zhang 等人<sup>[9]</sup>提出一种基于分解的多目标优化算法(A Multiobjective Evolutionary Algorithm Based on Decomposition, MOEA/D),该算法在目标空间中生成均匀的权重向量,将多目标问题分解为多个单目标子问题,使个体沿着权重向量的方向向 Pareto 前沿收敛,从而得

到均匀分布的 Pareto 解集.

随着目标维数的增加,传统多目标优化算法的缺陷<sup>[10]</sup>逐渐显现:(1)种群中非支配个体的比例迅速上升,传统 Pareto 支配关系对优秀解的选择压力衰减严重;(2)逼近整个 Pareto 前沿所需的解的个数呈指数级增长,想要获得规模巨大并覆盖高维目标空间中的整个 Pareto 前沿是极其困难的;(3)在高维目标优化问题中,算法的收敛性和多样性难以维持,即提升算法收敛性的同时会导致多样性的下降,反之亦然.

因此,有学者考虑将决策者的偏好融入到高维目标优化算法中.在实际情况下,决策者真正感兴趣的不是整个 Pareto 前沿,而是基于决策者偏好信息的某一局部区域 ROI(Region Of Interest).将搜索非支配解集的重点集中于 ROI 区域,不仅可以集中搜索资源,而且有助于决策者从大量解集中挑选满意解.例如:Fleming 等人<sup>[11]</sup>使用偏好链接的方法来解决目标空间中的优化问题,Deb 等人<sup>[12]</sup>将参考点的偏好信息结合到 NSGA-II 算法中,Molina 等人<sup>[13]</sup>提出一种基于 g-占优的高维目标优化算法,Said 等人<sup>[14]</sup>提出 r-占优的思想.国内学者也对偏好信息的表达方式进行了相应研究:邱飞岳等人<sup>[15]</sup>在 g-占优的基础上做进一步的改进,引入正、负偏好信息,提出双极偏好占优多目标进化算法;郑金华等人<sup>[16]</sup>提出一种基于权重迭代的偏好多目标分解算法(MOEA/D Based on User-Preference and Alternate Weight, MOEA/D-PRE).Wang 等人<sup>[17]</sup>提出一种多偏好驱动下的协同进化算法(Preference-Inspired Coevolutionary Algorithms for Many-Objective Optimization, PICEA-g),该算法通过将多个偏好与种群协同进化,保持偏好有效性的同时激励种群向更优的方向进化,但其每代进行的个体评价策略大大增加了算法复杂度. Deb 等人<sup>[18]</sup>提出基于偏好点的非支配排序算法(An Evolutionary Many-objective Optimization Algorithm Using Reference-point-based Nondominated Sorting Approach, NSGA-III),该算法结合空间分解的思想,设置偏好点引导种群进化,以双层模式设置较少的偏好点,获取相对均匀的解集.

然而上述融入决策者偏好的高维目标优化方法存在以下问题:

(1)基于传统 Pareto 支配关系的 g-占优、r-占优和双极偏好占优机制受参考点位置影响很大,当参考点位于某些极端位置时,很有可能造成算法

不收敛.

(2)在基于分解的多目标优化算法中引入决策者的偏好信息,能有效解决参考点位于极端位置时算法不收敛的问题,但是其算法收敛性和多样性在迭代过程中会产生冲突,具体体现在种群多样性在逼近 Pareto 前沿的过程中反复遭到破坏.

为解决以上问题,本文提出一种基于 APD 精英选择策略的偏好高维目标优化算法 G-RVEA,以提高算法在高维目标优化问题中的求解精度,主要研究内容如下:

(1)对 g-占优、r-占优、双极偏好占优和 MOEA/D-PRE 这 4 种偏好模型的理论原理进行分析,并指出各自的不足之处.

(2)将决策者偏好信息融入到基于分解的多目标优化算法中,提出偏好向量生成策略.通过该偏好向量生成策略形成的偏好区域能够有效处理偏好区域的边界问题,同时消除参考点位于可行域、不可行域和 Pareto 面等不同位置时对算法收敛性的影响.

(3)为平衡算法收敛性和多样性之间的冲突,引入角度惩罚距离(APD)机制,分析并改进该机制在算法搜索后期存在的缺陷,提出基于 APD 的精英选择策略.显著性检验结果表明,在 ZDT1-6 测试问题上,APD 精英选择策略所得解集整体质量显著提高.

(4)将提出的 G-RVEA 算法与 g-占优、r-占优、双极偏好占优和 MOEA/D-RPE 算法在 3 至 10 维目标的 DTLZ1-4 标准测试函数上进行性能对比测试,实验结果表明 G-RVEA 算法所求解集能够有效反映决策者的偏好信息,并且算法在偏好区域所求解集的整体质量更优.

## 2 基于参考点的偏好多目标进化算法研究现状分析

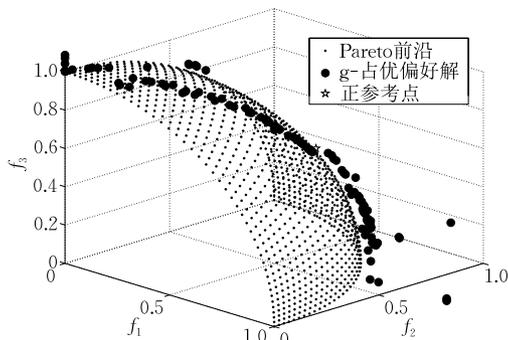
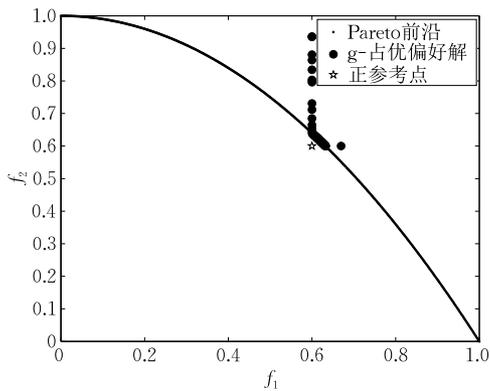
### 2.1 g-占优机制

Molina 等人提出一种名为 g-占优的基于参考点的多目标优化算法. g-占优的主要思想是根据决策者提供的参考点  $g$ , 定义偏好区域内的解占优非偏好区域的解,增加算法对非支配解的选择压力,使得算法搜索得到的 Pareto 非支配解集向参考点  $g$  附近的 ROI 区域收敛.

g-占优思想能非常方便地将决策者的偏好信息融入到多目标进化算法中,且该算法在 2 至 3 维目标优化问题中展现出非常良好的算法收敛性和多样

性. 然而,  $g$ -占优存在两个明显的缺点: 一是  $g$ -占优虽然增加偏好区域内的解占优非偏好区域的支配关系, 但同属于偏好区域内的解仍然是相互非支配的, 随着目标维数的增加,  $g$ -占优对偏好区域内非支配

解的选择压力显现疲态, 算法性能迅速下降; 二是当参考点靠近 Pareto 前沿附近, 尤其是位于 Pareto 前沿上时, 算法性能表现得非常不稳定, 甚至导致算法不收敛, 如图 1 所示.



(a) ZDT2

(b) DTLZ2

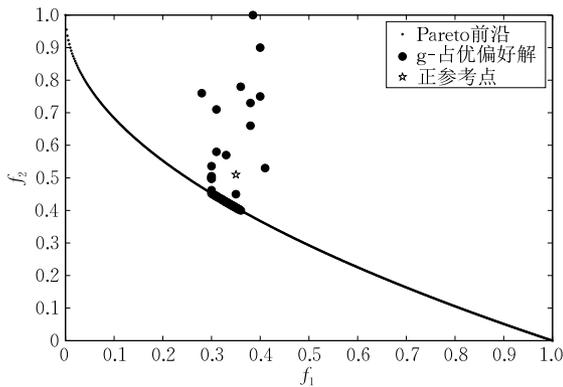
图 1 参考点位于前沿时,  $g$ -占优算法不收敛

### 2.2 r-占优机制

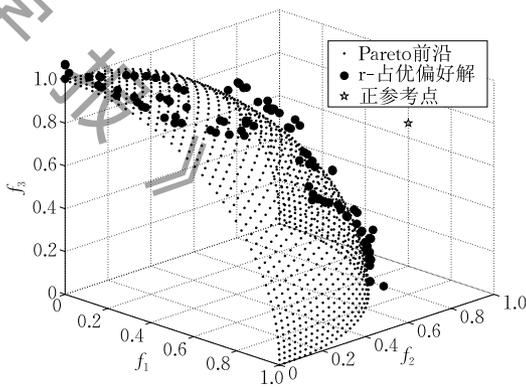
Said 等人<sup>[14]</sup>提出  $r$ -占优的思想, 以目标空间中个体和参考点的加权欧氏距离为基础, 在 Pareto 非支配解之间定义严格的偏序关系, 增强对优秀解的选择能力, 并通过  $\delta$  控制策略使得搜索过程逐渐向 ROI 区域收敛. 该占有机制与高维目标优化算法相结合后, 展现出良好的收敛性和多样性, 且随着目标

维数的增加, 依然能维持较好的算法性能.

当参考点位于可行域时, 算法搜索得到的偏好解集难以收敛到 Pareto 前沿, 极大地影响算法收敛性. 这是由于当参考点位于可行域时,  $r$ -占优机制将引导种群向参考点方向搜索而远离 Pareto 前沿, 导致种群的收敛速度放缓, 影响算法的搜索精度, 如图 2 所示.



(a) ZDT1



(b) DTLZ2

图 2 参考点位于可行域时,  $r$ -占优算法不稳定

### 2.3 双极偏好占优机制

邱飞岳、吴裕市等人<sup>[15]</sup>在  $g$ -占优的基础上做进一步的改进, 引入正、负偏好信息, 提出双级偏好占优多目标进化算法, 不但在 Pareto 非支配解之间定义严格的偏序关系, 而且同时考虑决策者的正负偏好信息, 提高算法对种群的选择压力, 引导种群向靠近正参考点同时远离负参考点的 ROI 区域收敛. 双极偏好占优机制在高维目标优化问题上的求解精度及

算法运行效率整体上优于  $g$ -占优机制和  $r$ -占优机制.

但是, 当决策者给出的正参考点远离 Pareto 前沿时, 双极偏好占优机制容易陷入局部最优而丢失部分最优解. 在图 3(a) 中, 双极偏好占优机制将搜索区域局限于上半部分, 丢失下半部分的偏好解; 而在图 3(b) 中, 双极偏好占优机制同样将搜索过程局限于靠近  $z$  轴的区域, 丢失靠近  $x$  轴和  $y$  轴的偏好解.

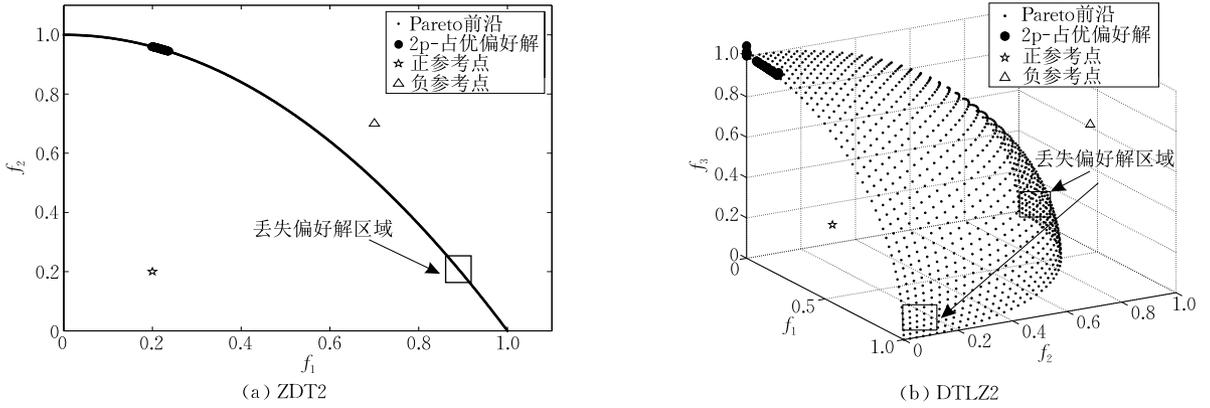


图 3 正参考点远离且负参考点靠近 Pareto 前沿时, 两极偏好占优丢失部分优秀解

### 2.4 MOEA/D-PRE

郑金华、喻果等人<sup>[16]</sup>将决策者的偏好信息融入到 MOEA/D 中, 提出一种基于权重迭代的偏好多目标算法 MOEA/D-PRE, 通过权重迭代方法获取一组均匀的权重向量并对偏好区域进行映射, 消除参考点位于目标空间极端位置时对算法性能的影响.

然而, MOEA/D-PRE 的收敛速度和种群多样性并不稳定. 在某些进化代数中, MOEA/D-PRE 搜索得到的偏好解集向 Pareto 前沿的收敛速度很慢; 当达到某一临界进化代数时, 搜索得到的偏好解集

会向 Pareto 前沿发生跃迁, 跃迁的同时种群的多样性会遭到严重破坏, 造成算法收敛性和多样性之间的冲突.

图 4 为 MOEA/D-PRE 在 2 维 DTLZ1 测试问题上迭代过程中的部分结果.  $gen=170\sim 290$  代时, MOEA/D-PRE 搜索偏好解集与 Pareto 前沿的距离几乎没有变化, 此时算法的搜索过程对收敛性几乎没有提升, 只是提高偏好解集在目标空间中的分布性;  $gen=290\sim 300$  代时种群朝着 Pareto 前沿的方向发生明显的跃迁, 算法搜索得到的偏好解集的

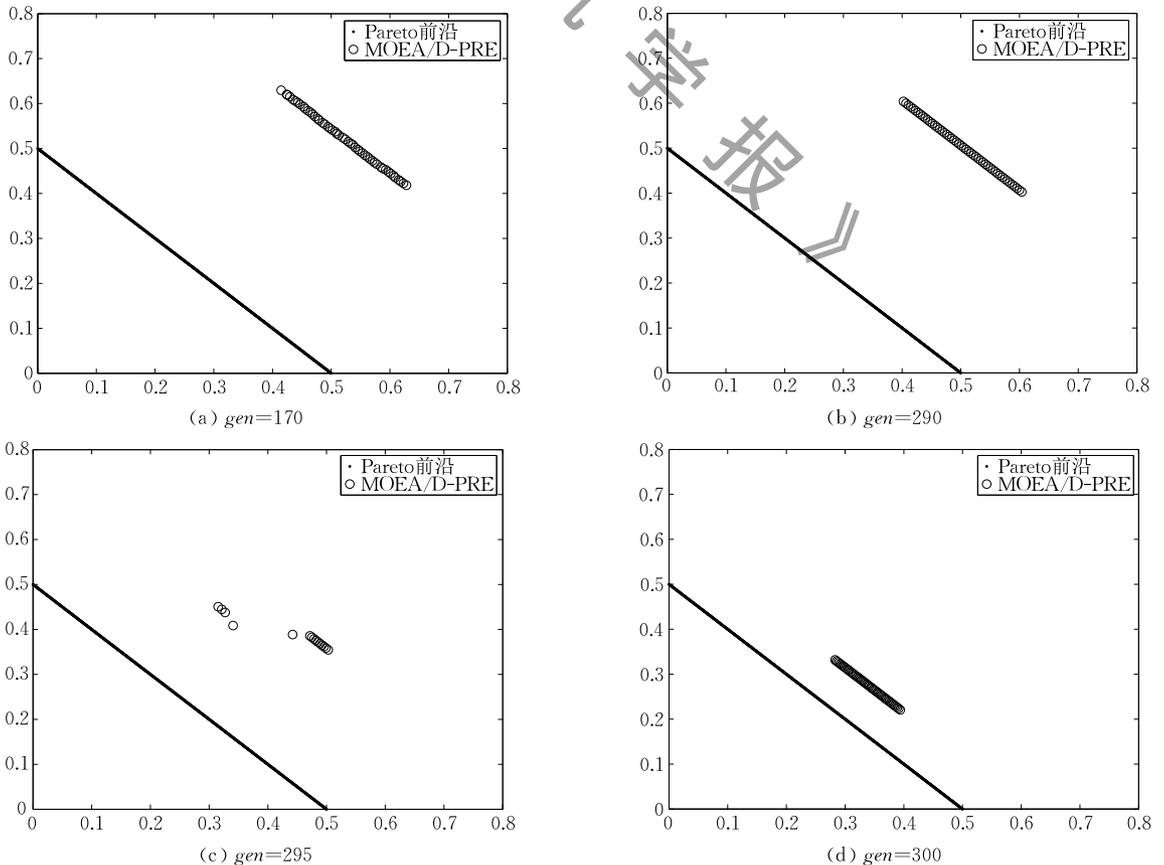


图 4  $gen=170\sim 300$  代, MOEA/D-PRE 在 2 维 DTLZ1 得到的解集

收敛性得到提高,但从  $gen=295$  代时的结果可以看出,在种群跃迁过程中,解集在目标空间中的分布性遭到破坏,换言之,在  $gen=170\sim 290$  代时算法提升的种群多样性在种群跃迁过程中反而下降; $gen=300$  代时,种群完成跃迁,此时算法再次着重于提升解集的分布性。

### 3 角度惩罚距离(APD)

#### 3.1 角度惩罚距离(APD)机制的原理

当种群  $P_i$  被划分为  $N$  个子种群:  $P_{t,1}, P_{t,2}, \dots, P_{t,N}$  后,需从各子种群中筛选出优秀解。然而,随着目标维数的增加,算法搜索优秀解的性能会出现不可避免的衰减,且算法收敛性和多样性往往难以在整个算法搜索过程中同时维持较高的水平。

Ishibuchi 等人<sup>[19]</sup>研究发现:提升算法收敛性时会不可避免地牺牲算法多样性,提升算法多样性时会不可避免地牺牲算法部分收敛性。

据此,为了平衡高维目标优化过程中算法收敛性和多样性的冲突,Cheng 等人<sup>[20]</sup>提出将 APD(角度惩罚距离)机制作为筛选个体的标准,如式(1)所示。其特点是在不同的搜索时期,算法性能的侧重点不同:在算法搜索前期,侧重算法收敛性;在算法搜索后期,侧重算法多样性。

$$d_{t,i,j} = (1 + P(\theta_{t,i,j})) \cdot \|F_{t,i}\| \quad (1)$$

式(1)中,  $d_{t,i,j}$  表示进化代数  $t$  时个体  $F_{t,i}$  的 APD 值,  $\theta_{t,i,j}$  表示个体  $F_{t,i}$  与其最近的偏好向量  $V_{t,j}$  之间夹角,  $\|F_{t,i}\|$  表示个体  $F_{t,i}$  与原点的欧几里得距离,  $P(\theta_{t,i,j})$  是关于  $\theta_{t,i,j}$  的惩罚函数,  $\theta_{t,i,j}$  越大,  $P(\theta_{t,i,j})$  的值也越大,其表达式如下:

$$P(\theta_{t,i,j}) = M \cdot \left(\frac{t}{t_{\max}}\right)^\alpha \cdot \frac{\theta_{t,i,j}}{\gamma_{V_{t,j}}} \quad (2)$$

$$\gamma_{V_{t,j}} = \min_{i \in \{1, \dots, N\}, i \neq j} \langle V_{t,i}, V_{t,j} \rangle \quad (3)$$

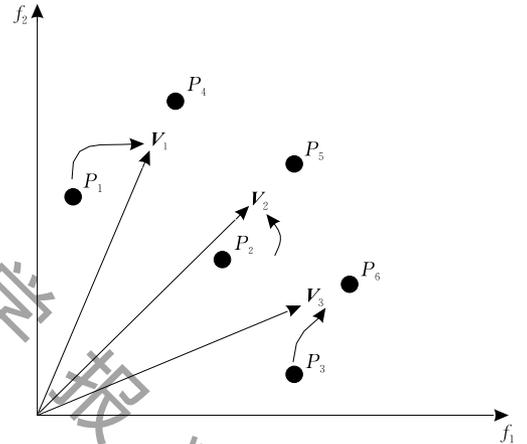
其中,  $M$  表示目标维数,也是惩罚系数;  $N$  表示偏好向量个数;  $t_{\max}$  表示最大进化代数;  $t$  表示当前进化代数;  $\alpha$  是事先定义的一个参数,用于控制  $P(\theta_{t,i,j})$  随当前进化代数  $t$  的变化率;  $\gamma_{V_{t,j}}$  表示偏好向量  $V_{t,j}$  与其他偏好向量之间的最小夹角。

从式(1)中可以看出,  $d_{t,i,j}$  与  $\theta_{t,i,j} \|F_{t,i}\|$  都是正相关的,个体  $F_{t,i}$  与其最近的偏好向量  $V_{t,j}$  之间夹角和与原点的欧几里得距离越小,其 APD 值也越小。

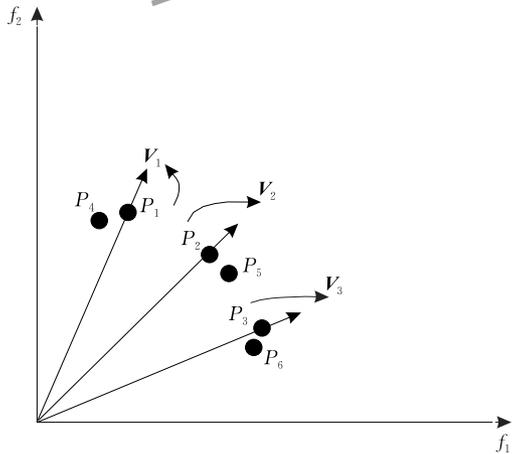
从式(2)中可以看出,在算法搜索前期( $t \ll t_{\max}$ ),  $P(\theta_{t,i,j}) \approx 0$ , 即  $d_{t,i,j} \approx \|F_{t,i}\|$ 。这意味着,算法主要通过比较子种群中个体  $F_{t,i}$  与目标空间原点的距离

对候选解进行筛选,使种群快速逼近 Pareto 前沿,从而提升算法收敛性。在图 5(a)中,个体  $F_1$  和  $F_4$  都被划分给偏好向量  $V_1$ ,由 APD 计算公式可知,由于  $d_1 \approx \|F_1\| < d_4 \approx \|F_4\|$ ,因此个体  $F_1$  更优。同理,  $F_2$  优于  $F_5$ ,  $F_3$  优于  $F_6$ ,保证搜索得到的解集快速向 Pareto 前沿推进。

而在算法搜索后期( $t \approx t_{\max}$ ),此时  $P(\theta_{t,i,j}) \approx M \cdot (\theta_{t,i,j} / \gamma_{V_{t,j}})$ ,即与偏好向量  $V_{t,j}$  之间角度大的个体  $F_{t,i}$  将被惩罚,惩罚系数为  $M$ 。这意味着,在搜索后期,算法保留子种群中与偏好向量  $V_{t,j}$  角度小的个体  $F_{t,i}$ ,从而维持算法多样性。在图 5(b)中,同样是个体  $F_1$  和  $F_4$  被划分给偏好向量  $V_1$ ,由式(2)可知,由于个体  $F_1$  与偏好向量  $V_1$  之间的夹角更大,受到更大的惩罚,因此  $d_4 < d_1$ ,即个体  $F_4$  更优。同理,  $F_5$  优于  $F_2$ ,  $F_6$  优于  $F_3$ ,保证搜索得到的个体均匀地分布在偏好向量的方向上,从而维持种群的多样性。



(a) 算法搜索前期



(b) 算法搜索后期

图 5 角度惩罚距离(APD)机制示意图

#### 3.2 角度惩罚距离(APD)机制的缺陷

角度惩罚距离(APD)机制可以缓解高维目标优化问题中算法收敛性和多样性之间的冲突。然

而在算法搜索后期, APD 机制为维持种群的多样性, 牺牲种群的收敛性, 具体体现在以下两方面: (1) APD 机制会选择 Pareto 被支配的个体进入下一代, 影响算法的收敛性; (2) APD 机制会淘汰部分父代中优秀解, 导致种群的退化。

在图 5(b) 中, 假设个体  $F_4$ 、 $F_5$ 、 $F_6$  是父代, 经过遗传操作得到子代  $F_1$ 、 $F_2$ 、 $F_3$ , 通过父子代合并、种群划分后得到子种群  $P_1 = \{F_1, F_4\}$ ,  $P_2 = \{F_2, F_5\}$ ,  $P_3 = \{F_3, F_6\}$ 。从 APD 机制分析: 个体  $F_4$  优于  $F_1$ ,  $F_5$  优于  $F_2$ ,  $F_6$  优于  $F_3$ , 因此个体  $F_4$ 、 $F_5$  和  $F_6$  进入下一代。但从传统 Pareto 支配关系分析, 个体  $F_1$  支配  $F_4$ ,  $F_3$  支配  $F_6$ , 即应选择个体  $F_1$ 、 $F_5$  和  $F_3$  进入下一代。换言之, 在算法搜索后期, 角度惩罚距离 (APD) 机制可能会淘汰 Pareto 非支配解, 而选择 Pareto 被支配解进入下一代。本文 5.3 节仿真实验统计了 APD 机制在 ZDT1-6 系列测试函数上的 GD 和 SP 指标, 验证 APD 机制搜索缺陷。

## 4 G-RVEA 算法

### 4.1 基于 APD 精英选择策略的偏好高维目标优化算法 G-RVEA

本文提出的 G-RVEA 算法主要包括偏好向量生成策略、种群划分策略和 APD 精英选择策略三大部分, 其具体步骤如下:

1. 设置基本参数, 并初始化种群;
2. 通过决策者给定参考点  $g$  和偏好区域半径参数  $\epsilon$ , 生成特定的偏好区域;
3. 对父代种群进行交叉变异, 产生子代种群, 并合并父子代种群;
4. 将种群中的个体划分给最近的偏好向量, 生成子种群;
5. 判断子种群个数是否达到种群规模  $N$ 。若是, 则对各子种群中的个体进行 Pareto 非支配排序, 剔除被支配的个体, 再对剩下的 Pareto 非支配个体计算 APD 值, 并选择子种群中 APD 值最小的个体进入下一代;
6. 判断是否达到最大进化代数。若是, 则输出当前种群作为最终结果; 否则转步 3。

在算法搜索前期, G-RVEA 算法侧重种群收敛性, 通过增加对解个体离目标原点距离的选择压力, 迫使种群快速向 Pareto 前沿收敛; 在算法搜索后期, G-RVEA 算法侧重种群多样性, 通过对解个体与偏好向量之间的夹角进行惩罚, 维持种群在高维目标空间下的均匀分布。G-RVEA 的算法流程如算法 1 所示。

### 算法 1. G-RVEA 算法.

输入: 种群规模  $N$ , 最大进化代数  $t_{\max}$ , 参考点  $g$ , 偏好区域参数  $\epsilon$ , APD 参数  $\alpha$ , 交叉概率  $P_m$ , 变异概率  $P_c$

输出: 最终种群  $P_{t=gen}$

1.  $V = \text{preference-vector-generation}(N, g, \epsilon)$
2. FOR  $t=1$  TO  $t_{\max}$
3.  $Q_t = \text{genetic-operator}(P_t, P_m, P_c)$
4.  $P_t = P_t \cup Q_t$
5.  $P_{t,i} = \text{population-division}(P_t, V)$
6.  $P_{t+1} = \text{APD-elitism-selection}(t, \alpha, V, P_{t,i})$
7. END FOR

### 4.2 偏好向量生成策略

基于传统 Pareto 支配关系的偏好占优机制受参考点位置的影响很大, 当参考点位于某些极端位置时, 很有可能造成算法不收敛。此外, 尽管这些占有机制在非支配解中确定更为严格的偏序关系, 但随着目标维数的增加, 搜索得到的非支配解比例仍然很高, 极大地影响算法性能。

因此, 本文将决策者偏好信息融入到基于分解的多目标优化算法中, 提出一种基于等比缩放的偏好向量生成策略。该策略通过决策者给定参考点  $g$  和偏好区域半径参数  $\epsilon$ , 将目标空间中均匀分布的权重向量按比例缩放, 形成偏好向量。

权重向量采用式 (4)~(6) 所示的单格子点法<sup>[21]</sup>生成, 式中  $i=1, 2, \dots, N$  表示权重向量的个数,  $M$  表示目标维度,  $H$  是用户定义的正整数,  $W_i$  表示均匀分布的权重向量。由单格子法的性质可知, 给定目标维度  $M$  和正整数  $H$ , 权重向量的个数由公式  $N = C_{H+M-1}^{M-1}$  确定。

$$u_i = (u_i^1, u_i^2, \dots, u_i^N) \quad (4)$$

$$u_i^j \in \left\{ \frac{0}{H}, \frac{1}{H}, \dots, \frac{H}{H} \right\} \quad (5)$$

$$W_i = \frac{u_i}{\|u_i\|} \quad (6)$$

图 6 表示在三维目标空间中,  $H=3$  时产生的权重向量, 此时权重向量个数  $N = C_{3+3-1}^{3-1} = 10$ 。从图中可以看出, 通过单格子点生成的权重向量在目标空间中是均匀分布的。然后将决策者给定的参考点  $g$  与目标空间原点  $O$  的连线方向定为参考方向, 参考向量  $OG$  与超平面  $f_1 + f_2 + \dots + f_m = 1$  交于映射点  $G$ , 如图 7 所示。

接着, 根据决策者给定的偏好区域半径参数  $\epsilon$ , 将权重向量按式 (7) 进行比例缩放, 生成一组均匀分布的偏好向量  $V = \{V_1, V_2, \dots, V_N\}$ 。

$$V_i = G_i + (W_i - G_i) \cdot \epsilon_i \quad (7)$$

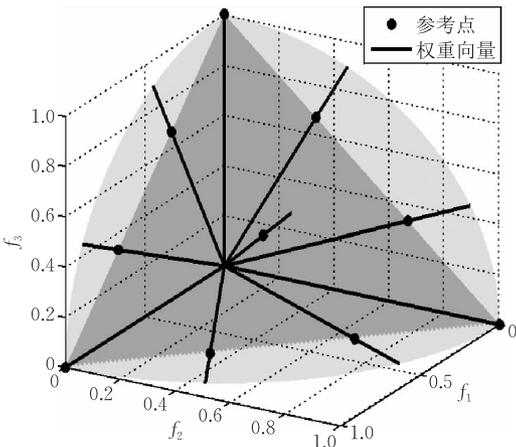


图 6 权重向量示意图

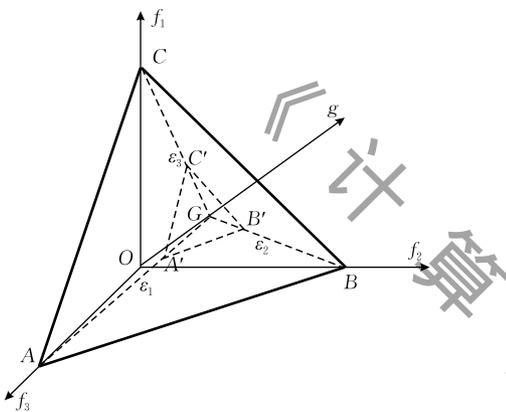


图 7 映射点示意图

式中,  $W_i$  是权重向量  $W$  在第  $i$  维目标上的函数值,  $G_i$  是映射点  $G$  在第  $i$  维目标上的值,  $\epsilon_i$  是参数  $\epsilon$  在第  $i$  维目标上的值且  $\epsilon_i = [0, 1]$ .  $V_i$  表示缩放后生成的偏好向量在第  $i$  维目标上的值. 偏好向量生成策略的算法流程如算法 2 所示.

**算法 2.** 偏好向量生成策略.

输入: 参考点  $g$ 、偏好区域半径参数  $\epsilon = (\epsilon_1, \epsilon_2, \dots, \epsilon_m)$ 、

参数  $H$

输出: 偏好向量  $V = \{V_1, V_2, \dots, V_N\}$

1.  $W = \text{weight-vector-Initialization}(H)$
2.  $G = \{(G_1, G_2, \dots, G_m) | G_1 + G_2 + \dots + G_m = 1, G_1/g_1 = G_2/g_2 = \dots = G_m/g_m\}$
3. FOR  $i=1$  TO  $N$
4.  $V_i = G_i + (W_i - G_i) \cdot \epsilon_i$
5. END FOR

该偏好向量生成策略有以下优点:

(1) 消除算法收敛性对参考点位置的敏感性. 在  $\epsilon$  相同的情况下, 参考向量  $OG$  方向上的不同参考点  $g$ , 产生的偏好向量是相同的, 即产生相同的偏好区域, 消除参考点位于可行域、不可行域以及 Pareto 面上时, 对算法收敛性的影响.

(2) 偏好区域的均匀性. 通过单格子点法产生的权重向量比例缩放形成得到偏好向量十分均匀, 具有良好的分布性. 决策者可以通过对参数  $\epsilon_i = [0, 1]$  的设置来调整偏好区域的大小, 如图 8 所示.

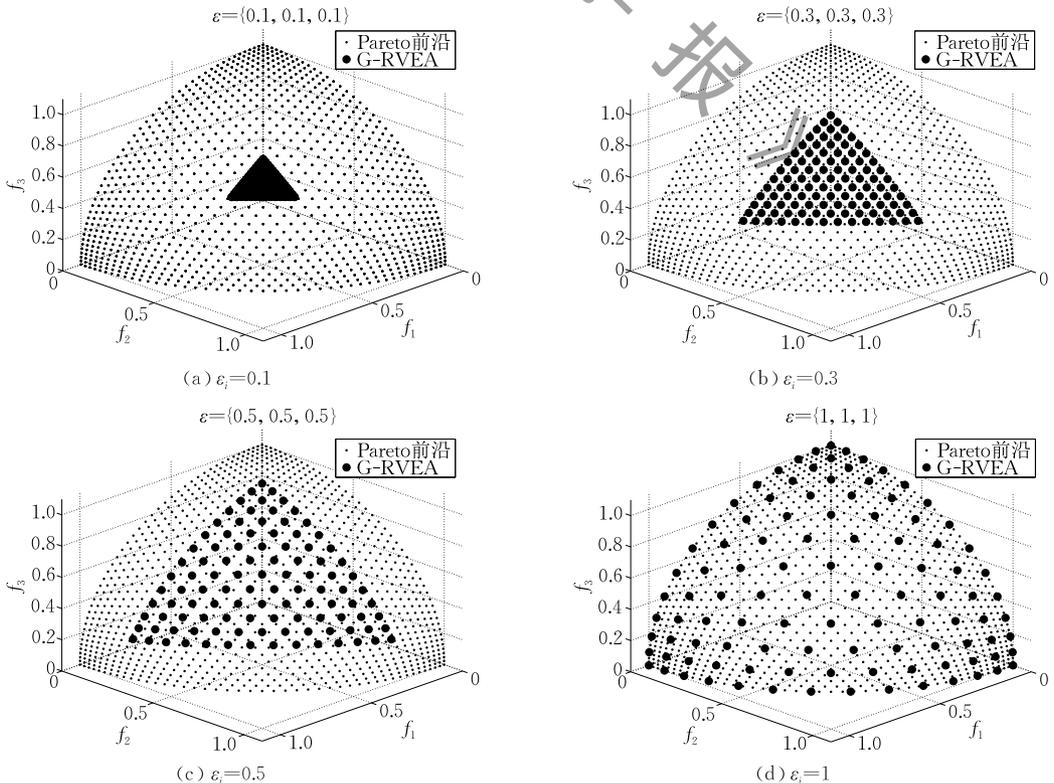


图 8 参数  $\epsilon_i$  不同大小下生成的偏好区域

(3) 搜索范围的灵活性. 通过对参数  $\epsilon_i$  的设置, 可以使算法对不同维度的目标上分配不同权重的搜索范围, 帮助决策者对目标偏好进行二次控制, 如图 9 所示.

(4) 偏好向量的有效性. 当参考点  $g$  位于极端位置时, 生成的偏好向量都能保证位于目标空间的第一象限内, 即有效地处理偏好区域的边界问题, 如图 10 所示.

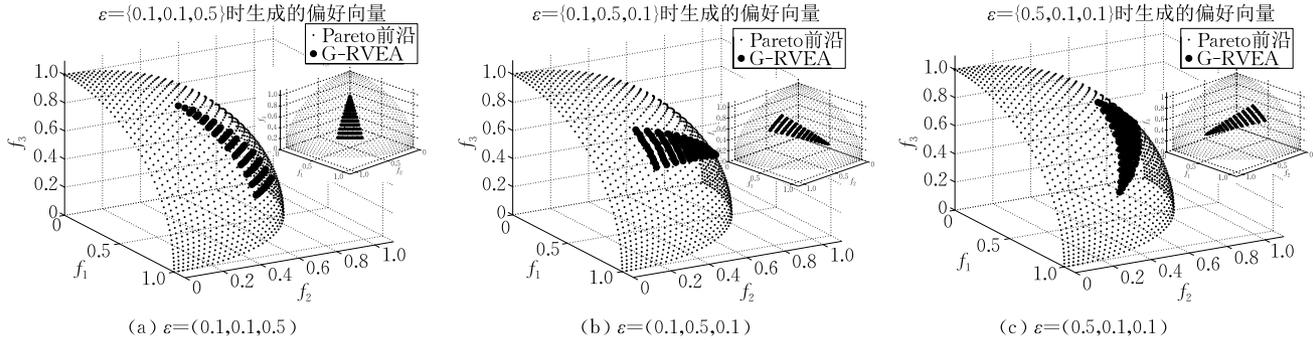


图 9 决策者对目标偏好的二次控制

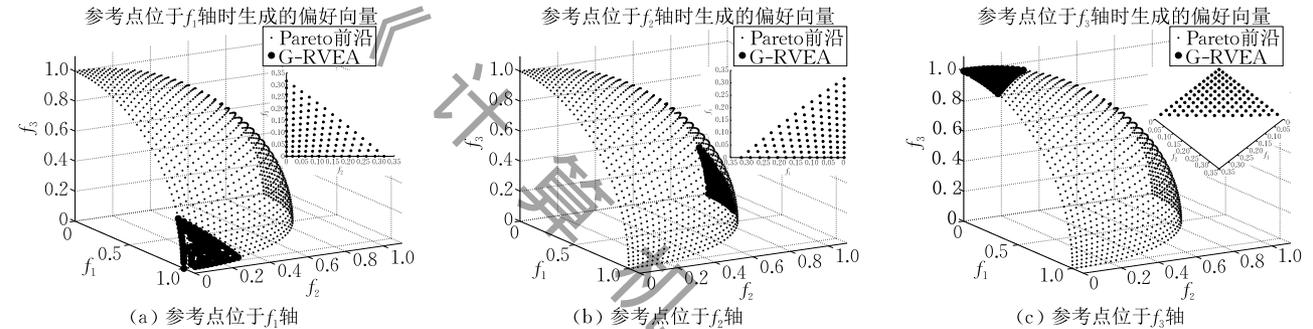


图 10 参考点位于边界时生成的偏好区域

### 4.3 种群划分和精英策略

种群划分策略通过比较目标空间中目标向量与各偏好向量之间的夹角大小, 将种群  $P_t$  划分为  $N$  个子种群:  $P_{t,1}, P_{t,2}, \dots, P_{t,N}$ . 图 11 所示的目标空间中有 3 个偏好向量  $\mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3$  和 3 个目标向量  $\mathbf{F}_1, \mathbf{F}_2, \mathbf{F}_3$ . 由于目标向量  $\mathbf{F}_1$  与偏好向量  $\mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3$  的最小夹角为  $\theta_{1,1}$ , 因此个体  $\mathbf{F}_1$  被划分给偏好向量  $\mathbf{V}_1$ . 同理, 个体  $\mathbf{F}_2, \mathbf{F}_3$  被划分给偏好向量  $\mathbf{V}_2$ . 因此, 通过种群划分策略划分的子种群如下:

$$P_1 = \{\mathbf{F}_1\}, P_2 = \{\mathbf{F}_2, \mathbf{F}_3\}, P_3 = \{\phi\}.$$

从图 11 中可以看出, 在种群划分的过程中, 有可能出现参考向量没有被划分个体的情况如  $P_3 = \{\phi\}$ . 因此, 本文选择父代与子代合并的精英策略作为后代选择机制. 精英策略可以避免父代中的优秀解在迭代过程中被丢失, 从而保证在算法后期每个参考向量都有个体划分. 种群划分策略的算法流程如算法 3 所示.

#### 算法 3. 种群划分策略.

输入: 种群  $P_t$ 、偏好向量  $\mathbf{V} = \{\mathbf{V}_1, \mathbf{V}_2, \dots, \mathbf{V}_N\}$

输出: 子种群  $\{P_{t,1}, P_{t,2}, \dots, P_{t,N}\}$

1. FOR  $i=1$  TO  $N$
2.   FOR  $j=1$  TO  $N$
3.      $\mathbf{F}_i = \text{Function-value}(P_t)$
4.      $\cos(\theta_{i,j}) = \langle \mathbf{F}_{t,i}, \mathbf{V}_{t,j} \rangle$
5.   END FOR
6. END FOR
7. FOR  $i=1$  TO  $N$
8.    $k = \max(\cos(\theta_{i,j}))$
9.    $P_{t,k} = P_{t,k} \cup \{\mathbf{F}_{t,i}\}$
10. END FOR

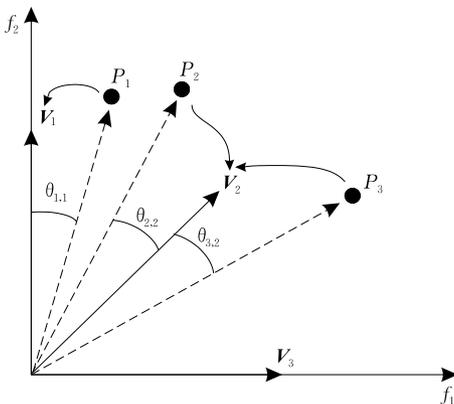


图 11 种群划分策略示意图

#### 4.4 改进后的 APD 精英选择策略

为弥补 3.2 节中提出的角度惩罚距离 (APD) 机制的缺陷, 本文对 APD 机制进行如下改进:

(1) 对种群划分后产生各子种群中的个体先进行 Pareto 非支配排序, 剔除被支配的个体, 然后对子种群中的 Pareto 非支配个体计算 APD 值, 选择 APD 值最小的个体进入下一代。

(2) 通过设置参数  $n$  的阈值, 确定在算法中引入 Pareto 非支配排序的时机, 其中  $n$  表示进化代数  $t$  时算法所求子种群的个数。

在搜索初期, 个体在目标空间中的分布是随机的, 通过种群划分策略得到的部分子种群为空集 (如 4.3 节所述), 导致通过 APD 机制筛选出的个体数量小于初始设置的种群规模。若此时引入 Pareto 非支配排序, 会增大个体数量达到初始设置种群规模所需的迭代次数, 导致算法收敛性放缓。因此, 本文设置该阈值  $n$  的值与种群规模  $N$  的大小相一致, 即当算法求得子种群个数达到初始设置的种群规模时, 再引入 Pareto 非支配排序。

需要注意的是, 本文引入的 Pareto 非支配排序与 Deb 等人提出的快速非支配排序有所不同。Deb 等人提出的快速非支配排序需要对种群中所有个体进行 Pareto 占优比较, 并根据个体的受支配程度对所有个体进行分层; 而本文引入的 Pareto 非支配排序只需对各子种群中的个体进行排序比较, 并筛选出 Pareto 支配程度最高的个体, 其计算复杂度低于前者。显著性检验结果表明, 与 APD 机制相比, 改进后 APD 精英选择策略所得解集整体质量显著提高。其算法流程如算法 4 所示。

##### 算法 4. APD 精英选择策略。

输入: 子种群  $\{P_{t,1}, P_{t,2}, \dots, P_{t,N}\}$ 、偏好向量  $\mathbf{V} = \{\mathbf{V}_1, \mathbf{V}_2, \dots, \mathbf{V}_N\}$ 、当前进化代数  $t$ 、最大进化代数  $t_{\max}$ 、APD 参数  $\alpha$

输出: 下一代种群  $P_{t+1}$

1. FOR  $j=1$  TO  $N$
2. IF  $N=n$
3.  $P_{t,j} = \text{Nondominated-sort}(P_t)$
4. END
5. FOR  $i=1$  TO  $\|P_{t,j}\|$
6.  $d_{t,i,j} = (1 + P(\theta_{t,i,j})) * \|\mathbf{F}_{t,i}\|$
7. END FOR
8. END FOR
9. FOR  $j=1$  TO  $N$
10.  $k = \min(d_{t,i,j})$
11.  $P_{t+1} = P_{t+1} \cup \{P_{t,k}\}$
12. END FOR

该 APD 精英选择策略有以下优点:

(1) 有效分配算法资源, 平衡高维目标优化过程中算法收敛性和多样性之间的冲突: 搜索前期, 算法侧重种群的收敛性, 以个体离目标空间原点的距离为筛选标准, 迫使种群快速逼近 Pareto 前沿, 极大地节省在高维目标空间中算法初期的计算成本; 搜索后期, 算法侧重种群的多样性, 通过放缓种群朝 Pareto 前沿的收敛速度, 剔除 Pareto 被支配解并对偏离偏好向量方向的个体进行惩罚, 在保证算法收敛性的基础上维持高维目标空间中种群的高多样性和解集的高均匀性。

(2) 解决种群退化问题, 使种群更快速地逼近 Pareto 前沿, 加快算法收敛速度, 从而使得算法后期有更多的计算资源用于提升种群的多样性。

(3) 将多目标问题分解为多个单目标子问题, 迭代过程中算法只需要对各子种群中的个体进行筛选, 提高算法搜索效率。

#### 4.5 G-RVEA 的时间复杂度

从算法 1 中可以看出, 除了交叉变异等遗传操作以外, G-RVEA 的时间复杂度主要源于种群划分策略和 APD 精英选择策略。种群划分策略的时间复杂度为  $O(MN^2)$ , 其中  $M$  表示目标维度,  $N$  表示种群规模。而 APD 精英选择策略的时间复杂度由 APD 机制和 Pareto 非支配排序两部分构成: (1) 在搜索初期, 种群划分后产生的子种群个数较少, 每个子种群中的个体数目较高, APD 机制的时间复杂度为  $O(M \times 2N)$ , 并且此时还没有引入 Pareto 非支配排序, 因此 APD 精英选择策略的时间复杂度为  $O(MN)$ ; (2) 在搜索后期, 种群划分后产生的子种群个数为  $N$ , 个体数被均匀地分配到各子种群, 此时引入的 Pareto 非支配排序的时间复杂度为  $O(M \times 2N/N)$ , 因此 APD 精英选择策略的时间复杂度还是  $O(MN)$ 。综上所述, G-RVEA 的时间复杂度为  $O(MN^2)$ 。

## 5 仿真实验及分析

### 5.1 参数设置

本实验测试环境为 Inter (R) Core (TM) i5-5200U CPU @ 2.20 GHz 4.0 GB。采用模拟二进制交叉和多项式变异, 交叉分布指数  $\eta_c = 30$ , 变异分布指数  $\eta_m = 20$ , 交叉概率  $P_c = 0.99$ , 变异概率  $P_m = 1/n$  ( $n$  为决策变量的个数)。参考点  $g_i = 0.2$ 。所有对比算法的参数  $H$  和种群规模  $N$  的设置如表 1 所示。

表 1 目标维度  $M$ 、参数  $H$  与种群规模  $N$ 

目标维度 $M$	参数 $H$	种群规模 $N$
2	99	100
3	13	105
4	7	120
6	4	132
8	3	156
10	3	275

需要注意的是,随着目标维数的增加,逼近整个 Pareto 前沿所需的解的个数呈指数级增长,因此种群规模  $N$  也随之增大. 测试对象为 ZDT 系列测试函数<sup>[22]</sup>和 DTLZ1-4 系列测试函数<sup>[23]</sup>. ZDT 系列测试函数的最大进化代数为  $gen\_max = 300$ , DTLZ2 和 DTLZ4 测试函数的最大进化代数为  $gen\_max = 500$ , DTLZ1 和 DTLZ3 测试函数为最大进化代数为  $gen\_max = 1000$ . MOEA/D-PRE 和 G-RVEA 的偏好区域半径参数在每一维上的值均设置为  $\epsilon_i = 0.1$ . “r-占优”中的参数  $\delta = 0.1$ ; “双极偏好占优”中的参数  $\delta = 0.0001$ ; MOEA/D-PRE 中的领域大小参数  $T = 20$ , 参数  $\theta = 5$ ; G-RVEA 中参数  $\alpha = 0.1$ . 为消除负参考点对偏好区域的影响,将其设置为距离目标原点非常远的点  $g_i^- = 100$ .

## 5.2 算法性能评价指标

仿真实验中,本文选用 4 个相关指标衡量算法在不同方面的性能.  $GD$  指标<sup>[24]</sup>用于衡量算法的收敛性,  $SP$  指标<sup>[24]</sup>用于衡量解集分布的均匀性,  $\epsilonpsilon$  指标<sup>[25]</sup>用于比较两个解集之间的受支配程度,  $HV$  指标<sup>[26]</sup>用于衡量算法的综合性能.

$GD$  指标是一种收敛性指标,表示算法得到的 Pareto 近似解集与 Pareto 前沿的平均距离,计算公式如下所示. 其中,  $n$  表示 Pareto 前沿中选取的标准点的个数,  $d_i$  表示算法得到的第  $i$  个个体与 Pareto 前沿的最小欧几里得距离.  $GD$  指标的值越小,算法得到的 Pareto 近似解集越逼近理论上真正的 Pareto 前沿.

$$GD = \frac{1}{n} \left( \sum_{i=1}^n d_i^2 \right)^{\frac{1}{2}} \quad (8)$$

$SP$  指标是一种均匀性指标,表示算法得到的 Pareto 近似解集在目标空间中的分布情况,计算公式如下所示. 其中  $n$  表示种群规模,  $d_i$  表示算法得到的第  $i$  个个体与其最近个体之间的欧几里得距离.  $SP$  指标的值越小,算法得到的 Pareto 近似解集在目标空间中的分布越均匀.

$$Spacing = \frac{\left[ \frac{1}{n} \sum_{i=1}^n (d_i - \bar{d})^2 \right]^{\frac{1}{2}}}{\bar{d}} \quad (9)$$

$\epsilonpsilon$  指标也是一种收敛性指标,用于比较两个解集之间的受支配程度. 给定任意两个解集  $A$  和  $B$ , 通过  $\epsilonpsilon$  指标计算公式可得到一对输出  $(I_A, I_B)$ , 计算公式如下所示:

$$I_A = \text{Inf} \{ \forall z^2 \in B, \exists z^1 \in A: z^1 <_{\epsilon} z^2 \} \quad (10)$$

$$I_B = \text{Inf} \{ \forall z^2 \in A, \exists z^1 \in B: z^1 <_{\epsilon} z^2 \} \quad (11)$$

其中,  $z^1 <_{\epsilon} z^2$  当且仅当  $\forall 1 \leq i \leq n: z_i^1 \leq \epsilon \cdot z_i^2$ .  $Z^1$  表示解集  $A$  中的任意一个目标向量,  $Z^2$  表示解集  $B$  中的任意一个目标向量,  $i$  表示目标维度. 因此, 输出  $(I_A, I_B)$  的计算公式等价于:

$$I_A = \max_{z^2 \in B} \min_{z^1 \in A} \max_{1 \leq i \leq n} \frac{z_i^1}{z_i^2} \quad (12)$$

$$I_B = \max_{z^2 \in A} \min_{z^1 \in B} \max_{1 \leq i \leq n} \frac{z_i^1}{z_i^2} \quad (13)$$

上式中,假定所求问题为最小化问题,对于输出对  $(I_A \leq 0, I_B \geq 0)$  表示解集  $A$  严格优于解集  $B$ ;  $(I_A \geq 0, I_B \geq 0)$  表示解集  $A$  和  $B$  之间无法比较,但如果  $I_A$  小于  $I_B$ , 可以认为解集  $A$  弱优于解集  $B$ ; 同理,  $(I_A \leq 0, I_B \leq 0)$  且  $I_A$  小于  $I_B$ , 则认为解集  $A$  弱优于解集  $B$ .

$HV$  指标也被称为 Lebesgue 测度,是一种能够同时衡量算法收敛性和多样性的综合性指标,表示非支配解集覆盖的目标空间区域大小,计算公式如下所示. 其中  $\Lambda$  即为 Lebesgue 测度,  $P_{front}$  表示所求解集,  $x_{ref}$  表示参照点.  $HV$  指标的值越大,算法求得解集质量越高.

$$HV(P_{front}) = \Lambda \left( \bigcup_{p \in P_{front}} \{x | p < x < x_{ref}\} \right) \quad (14)$$

## 5.3 APD 精英选择策略改进前后性能对比

为验证改进后 APD 精英选择策略的算法性能,测试其在 ZDT 系列函数上的  $GD$ 、 $SP$  以及  $HV$  指标,分别衡量算法收敛性、解集均匀性以及算法的综合性能,对每个测试函数独立运行 20 次并取平均值作为最终测试值,并用 T-test 检验改进前后算法性能提升的显著性,其中“+”、“-”和“=”分别表示所测指标值在显著水平为 5% 的情况下, APD 精英选择策略优于、劣于和无差别于 APD 机制,具体结果如表 2 所示.

仿真结果显示,在 ZDT1 测试问题上,改进后 APD 精英选择策略所求解集的收敛性 ( $GD$  指标) 优于 APD 机制,但其解集均匀性 ( $SP$  指标) 却劣于 APD 机制,导致改进前后算法所求解集的综合性能 ( $HV$  指标) 无明显差异. 该结果表明对于 ZDT1 这类 Pareto 前沿为连续凸的单模态简单测试问题, APD 机制所求解集已经十分逼近真实 Pareto 前沿,

表 2 改进前后算法  $GD$ 、 $SP$  以及  $HV$  指标对比结果

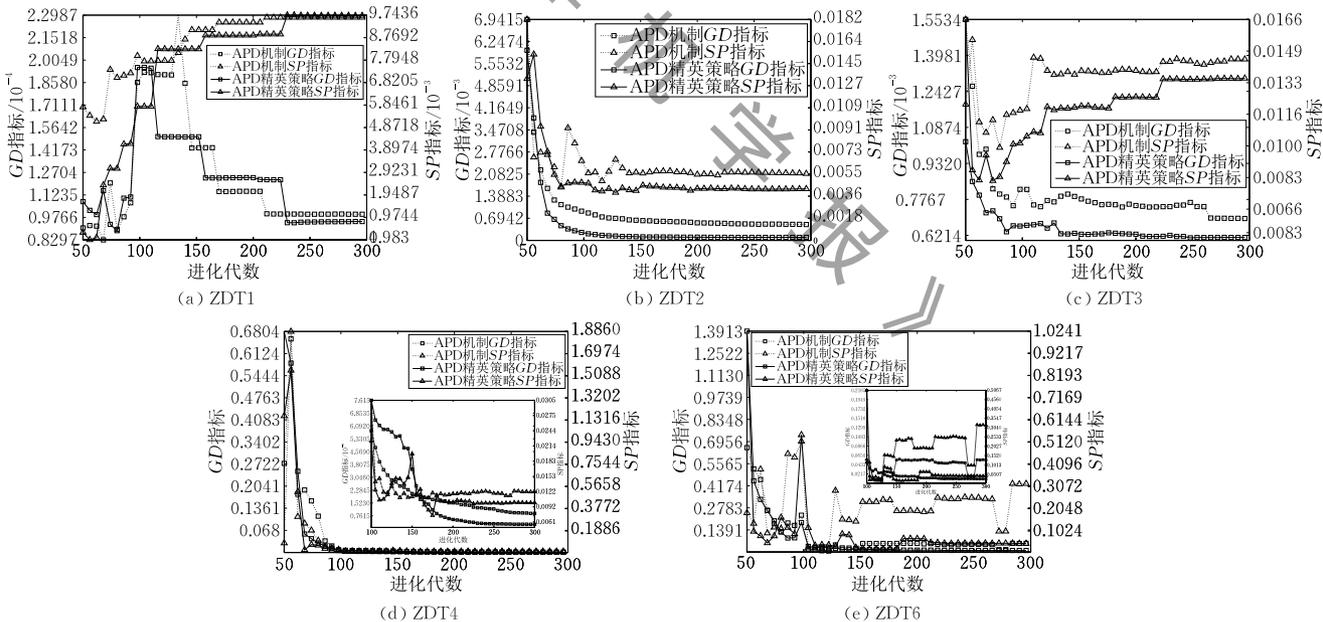
测试函数		APD 机制			APD 精英选择策略		
		$GD$	$SP$	$HV$	$GD$	$SP$	$HV$
ZDT1	均值	$9.02E-05$	<b><math>9.52E-03</math></b>	$6.61E-01$	<b><math>8.69E-05</math></b>	$9.70E-03$	<b><math>6.61E-01</math></b>
	方差	<b><math>1.80E-11</math></b>	<b><math>2.45E-08</math></b>	$4.27E-06$	$2.55E-11$	$2.98E-08$	<b><math>1.79E-06</math></b>
	T-test				+	-	=
ZDT2	均值	$4.95E-04$	$5.78E-03$	$3.22E-01$	<b><math>9.52E-05</math></b>	<b><math>4.13E-03</math></b>	<b><math>3.28E-01</math></b>
	方差	$1.29E-08$	$1.07E-06$	$3.24E-06$	<b><math>1.69E-11</math></b>	<b><math>3.69E-08</math></b>	<b><math>2.80E-06</math></b>
	T-test				+	+	+
ZDT3	均值	$7.17E-04$	$1.44E-02$	$5.23E-01$	<b><math>5.92E-04</math></b>	<b><math>1.37E-02</math></b>	<b><math>5.25E-01</math></b>
	方差	$1.87E-09$	$7.57E-08$	$5.71E-06$	<b><math>2.12E-12</math></b>	<b><math>1.90E-08</math></b>	<b><math>2.41E-06</math></b>
	T-test				+	+	+
ZDT4	均值	$5.71E-03$	$1.59E-02$	$6.33E-01$	<b><math>3.19E-04</math></b>	<b><math>9.66E-03</math></b>	<b><math>6.41E-01</math></b>
	方差	$2.03E-04$	$1.68E-04$	$1.79E-03$	<b><math>8.57E-09</math></b>	<b><math>3.31E-08</math></b>	<b><math>1.11E-03</math></b>
	T-test				+	+	+
ZDT6	均值	$3.55E-02$	$1.77E-01$	$5.12E-01$	<b><math>9.63E-03</math></b>	<b><math>4.21E-02</math></b>	<b><math>5.31E-01</math></b>
	方差	$1.05E-03$	$4.37E-02$	$1.16E-02$	<b><math>4.37E-05</math></b>	<b><math>1.58E-03</math></b>	<b><math>6.42E-03</math></b>
	T-test				+	+	+

APD 精英选择策略虽然能提升算法收敛性,但却牺牲了解集均匀性和算法多样性,整体提升效果不明显。

在 ZDT2-6 测试问题上,改进后的 APD 精英选择策略所求解集的收敛性( $GD$  指标)和解集均匀性( $SP$  指标)都优于 APD 机制,且综合性能( $HV$  指标)也更优。该结果表明对于 ZDT2(连续凹,单模态)、ZDT3(不连续,多模态)、ZDT4(连续凸,多模

态)、ZDT6(不连续凸,不均匀)等复杂测试问题,APD 精英选择策略能显著改善算法收敛性和多样性,提高算法所求解集的整体性能。

图 12(a)~(e)表示改进前后算法在 ZDT1-6 测试函数上  $GD$  和  $SP$  指标的变化曲线图,横坐标表示进化代数,左侧纵坐标表示  $GD$  指标值,右侧纵坐标表示  $SP$  指标值。

图 12 改进前后算法在 ZDT1-6 系列测试函数上  $GD$  和  $SP$  指标曲线图

在 ZDT1 测试问题上,进化代数  $t=[100, 200]$  时, $GD$  指标出现明显上升,且上升幅高达 200%,这表明在该进化时期内,APD 机制为了维持较高的多样性,过多地牺牲算法收敛性,使得种群长时间处于退化状态;而改进后的 APD 精英选择策略有效降低算法收敛性的上升幅度和搜索末期种群的最终收敛程度。

在 ZDT2、ZDT3 和 ZDT4 测试问题上,APD 机

制的  $GD$  指标在算法搜索过程呈现平稳下降的态势,这表明在该测试问题上 APD 机制的种群退化现象并不明显;APD 精英选择策略则进一步提高了算法的收敛程度,使得算法在此类问题上所求解集的  $GD$  指标得到了明显优化。

而在 ZDT6 测试问题上,进化代数  $t=[120, 260]$  时,种群同样出现退化现象, $GD$  和  $SP$  指标均

有明显上升;采用 APD 精英选择策略后,算法搜索过程中的种群退化问题得到改善, $GD$  指标呈平稳下降趋势, $SP$  指标仅有小幅度的上升,但整体指标值都有所降低。

除 ZDT1 测试问题以外,改进后的 APD 精英选择策略对算法收敛性和多样性都得到显著改善。这是因为 APD 精英选择策略剔除子种群中的被支配个体,避免种群退化现象,加快种群的收敛速度,使得算法有更多的计算资源用于维护种群的多样性和

解集分布的均匀性。

#### 5.4 偏好高维目标优化问题性能测试

为验证本文所提算法 G-RVEA 在偏好高维目标优化问题中的整体性能,采用  $\epsilon$ -SP 和运行时间  $t$  分别衡量 g-占优、r-占优、双极偏好占优、MOEA/D-PRE 和 G-RVEA 在 3~10 维目标优化问题上的算法收敛性、解集均匀性和时间复杂度。本文对每个测试函数独立运行 20 次并取平均值最为最终测试值,测试结果如表 3、表 4 和表 5 所示。

表 3 5 种算法在 3~10 维 DTLZ1-4 系列测试函数的  $\epsilon$ -SP 指标测试结果

测试函数	M	G-RVEA vs g-占优		G-RVEA vs r-占优		G-RVEA vs 双极偏好占优		G-RVEA vs MOEA/D-PRE	
		$I(A,B)$	$I(B,A)$	$I(A,B)$	$I(B,A)$	$I(A,B)$	$I(B,A)$	$I(A,B)$	$I(B,A)$
DTLZ1	3	<b>-3.4954</b>	4.4698	<b>-8.6234</b>	10.0459	<b>-1.4797</b>	7.1163	<b>0.0001</b>	0.0015
	4	<b>-1.3722</b>	5.4449	<b>-2.9292</b>	7.1898	<b>-1.1010</b>	3.7603	<b>0.0023</b>	0.0030
	6	<b>-2.5711</b>	9.0225	<b>-1.0950</b>	15.6787	<b>-0.1188</b>	2.8029	0.0065	<b>0.0031</b>
	8	<b>-0.0512</b>	14.4511	<b>-2.2058</b>	9.1810	<b>-0.5523</b>	4.7897	<b>0.0035</b>	0.0054
	10	<b>-2.6418</b>	17.5984	<b>-2.4792</b>	19.5654	<b>-0.7841</b>	9.8825	<b>0.0044</b>	0.0071
DTLZ2	3	<b>-0.0062</b>	0.0096	<b>-0.0303</b>	0.0082	<b>-0.0183</b>	0.0119	<b>0.0012</b>	0.0049
	4	<b>-0.0143</b>	0.0424	<b>-0.0506</b>	0.0736	<b>-0.0580</b>	0.0489	<b>0.0083</b>	0.0124
	6	<b>-0.1669</b>	0.5839	<b>0.3510</b>	0.8393	<b>-0.0914</b>	0.0992	0.0292	<b>0.0126</b>
	8	<b>-0.1821</b>	0.8005	<b>0.2591</b>	0.7050	<b>-0.2249</b>	0.1842	<b>0.0147</b>	0.0294
	10	<b>-0.2022</b>	0.8454	<b>0.1766</b>	0.6395	<b>0.2107</b>	0.2651	<b>0.0199</b>	0.0237
DTLZ3	3	<b>-34.9247</b>	43.5619	<b>-31.6390</b>	38.5384	<b>-2.1962</b>	41.8542	<b>-0.0042</b>	0.0105
	4	<b>-26.7964</b>	48.0572	<b>-7.7835</b>	44.4612	<b>-0.2450</b>	39.4312	<b>-0.0585</b>	0.0714
	6	<b>-13.5541</b>	139.2720	<b>-18.8990</b>	147.9110	<b>0.3249</b>	63.9551	<b>0.0263</b>	0.0334
	8	<b>-22.9063</b>	193.7560	<b>-34.2880</b>	165.5040	<b>-2.9877</b>	65.3551	<b>0.0184</b>	0.0282
	10	<b>-34.3586</b>	158.4940	<b>-34.9960</b>	171.0140	<b>0.2382</b>	53.1763	<b>0.0336</b>	0.0533
DTLZ4	3	<b>-0.0161</b>	0.0200	<b>-0.0193</b>	0.0158	0.5142	<b>0.1168</b>	<b>0.0010</b>	0.0047
	4	0.4433	<b>0.1697</b>	0.4429	<b>0.2727</b>	0.4438	<b>0.1967</b>	<b>0.0085</b>	0.0106
	6	<b>0.3555</b>	0.7015	<b>0.3569</b>	0.5040	<b>0.3614</b>	0.5304	<b>0.0197</b>	0.0327
	8	<b>0.3071</b>	0.9634	<b>0.2996</b>	1.0745	<b>0.3122</b>	0.5877	<b>0.0196</b>	0.0366
	10	<b>0.2678</b>	0.9844	<b>0.2695</b>	1.0757	<b>0.2816</b>	0.2869	0.2714	<b>0.0621</b>

表 4 5 种算法在 3~10 维 DTLZ1-4 系列测试函数的  $SP$  指标测试结果

测试函数	M	G-RVEA		g-占优		r-占优		双极偏好占优		MOEA/D-PRE	
		均值	方差	均值	方差	均值	方差	均值	方差	均值	方差
DTLZ1	3	<b>4.66E-05</b>	<b>1.90E-10</b>	5.68E-01	2.24E-01	3.58E+00	1.89E+01	4.77E-01	5.72E-01	2.78E-04	2.07E-08
	4	<b>9.50E-05</b>	<b>2.83E-09</b>	1.40E+00	2.10E-01	3.34E+00	2.00E+00	1.85E+00	3.14E-01	4.88E-03	8.19E-08
	6	<b>1.52E-03</b>	<b>2.61E-09</b>	3.57E+00	8.84E-01	1.60E+01	8.09E+00	3.26E+00	6.66E-01	1.03E-02	5.45E-08
	8	<b>3.25E-03</b>	<b>8.60E-09</b>	1.74E+01	7.34E+00	2.31E+01	1.29E+01	1.63E+01	2.42E+00	1.35E-02	6.54E-08
	10	<b>3.23E-03</b>	1.40E-07	8.45E+01	9.23E+00	9.24E+01	9.61E+00	6.31E+01	7.76E+00	3.27E-02	<b>6.88E-08</b>
DTLZ2	3	<b>4.52E-04</b>	<b>1.99E-08</b>	5.96E-02	9.82E-05	3.29E-02	1.17E-04	3.65E-02	3.05E-05	1.01E-03	1.53E-08
	4	<b>1.25E-03</b>	<b>1.18E-07</b>	1.33E-01	5.13E-04	9.21E-02	3.19E-04	5.48E-02	7.37E-04	2.94E-03	1.16E-07
	6	<b>7.46E-03</b>	<b>5.02E-07</b>	3.88E-01	2.06E-03	4.52E-01	3.76E-02	3.61E-01	7.26E-04	2.56E-02	4.45E-06
	8	<b>1.90E-02</b>	<b>1.05E-06</b>	5.15E-01	1.13E-03	9.24E-01	1.58E-02	2.40E-01	3.44E-04	6.82E-02	2.22E-05
	10	<b>2.05E-02</b>	<b>1.03E-06</b>	8.33E-01	4.63E-02	2.12E+00	6.52E-01	6.83E-01	8.93E-03	6.09E-02	3.93E-05
DTLZ3	3	<b>6.29E-04</b>	<b>3.43E-08</b>	3.76E+00	2.13E+01	1.49E+01	2.36E+02	3.85E+00	3.94E+00	1.77E-03	3.30E-06
	4	<b>1.17E-03</b>	<b>3.64E-07</b>	6.27E+00	2.13E+01	2.65E+01	1.12E+02	1.65E+01	5.79E+00	1.97E-02	2.75E-06
	6	<b>7.04E-03</b>	<b>3.47E-07</b>	2.72E+01	1.75E+02	8.83E+01	3.17E+02	1.94E+02	2.81E+01	4.56E-02	3.78E-06
	8	<b>1.94E-02</b>	<b>3.88E-06</b>	1.82E+02	3.79E+02	2.45E+02	1.06E+03	1.69E+02	2.40E+01	5.99E-02	7.15E-06
	10	<b>1.97E-02</b>	<b>1.00E-06</b>	9.68E+02	7.78E+02	1.54E+03	7.42E+04	8.46E+02	4.59E+01	8.09E-02	3.93E-06
DTLZ4	3	<b>7.09E-04</b>	<b>5.45E-08</b>	3.18E-02	4.98E-04	4.23E-02	2.04E-04	1.95E-02	3.61E-04	1.15E-03	3.32E-07
	4	<b>2.35E-03</b>	<b>1.99E-06</b>	8.53E-02	1.74E-03	8.88E-02	7.70E-04	7.30E-02	8.42E-04	1.39E-02	3.77E-06
	6	<b>1.34E-02</b>	9.85E-04	4.02E-01	4.06E-03	6.12E-01	6.11E-03	3.11E-01	1.01E-03	3.64E-02	<b>8.12E-07</b>
	8	<b>1.49E-02</b>	3.66E-05	6.31E-01	1.93E-03	9.60E-01	8.17E-03	4.10E-01	7.76E-03	4.76E-02	<b>1.12E-06</b>
	10	<b>2.10E-02</b>	3.28E-06	9.06E-01	3.36E-03	1.44E+00	3.88E-02	7.05E-01	1.82E-02	9.05E-02	<b>1.15E-06</b>

表 5 5 种算法在 3~10 维 DTLZ1-4 系列测试函数的运行时间

(单位:s)

测试函数	M	G-RVEA	g-占优	r-占优	双极偏好占优	MOEA/D-PRE
DTLZ1	3	<b>18.3840</b>	87.0210	1219.55	146.2460	161.0120
	4	<b>22.3114</b>	212.6850	1933.24	274.6560	193.4590
	6	<b>24.4842</b>	251.9910	2777.60	240.4980	202.3210
	8	<b>24.0361</b>	333.6820	4493.17	357.0460	251.7260
	10	<b>36.8317</b>	9148.6500	14652.70	9093.1200	457.4540
DTLZ2	3	<b>9.9104</b>	31.5949	1026.43	40.4103	88.1582
	4	<b>11.1884</b>	70.7693	1639.49	90.7187	111.8180
	6	<b>10.6051</b>	99.3096	2362.24	125.4810	118.1150
	8	<b>14.8489</b>	163.0030	3807.01	158.9220	136.6660
	10	<b>16.5495</b>	486.1350	12528.40	510.2310	204.4330
DTLZ3	3	<b>12.7637</b>	118.4200	1621.08	88.7036	186.6650
	4	<b>20.8584</b>	209.4040	1834.18	243.7960	214.4930
	6	<b>23.8092</b>	210.0540	2857.46	241.8520	240.0680
	8	<b>26.1452</b>	346.1890	4952.67	299.0010	265.9950
	10	<b>32.4113</b>	1007.4800	16788.80	883.8720	478.7820
DTLZ4	3	<b>9.1085</b>	40.4090	1149.30	54.0259	90.1672
	4	<b>9.8292</b>	69.4197	1508.39	94.6476	105.2520
	6	<b>11.4544</b>	83.9209	2475.10	115.5550	119.6910
	8	<b>15.0714</b>	160.6780	3901.27	173.2550	129.5340
	10	<b>27.1156</b>	519.2420	12931.40	496.2090	238.6710

#### 5.4.1 算法收敛性分析

从表 3 所示的  $\epsilon$  指标测试结果来看,在 DTLZ1 测试函数上,G-RVEA 严格优于 g-占优、r-占优和双极偏好占优;除 6 维目标以外,均弱优于 MOEA/D-PRE. 在 DTLZ2 测试函数上,G-RVEA 严格优于 g-占优;严格优于或弱优于 r-占优和双极偏好占优;除 6 维测试函数以外,均弱优于 MOEA/D-PRE. 在 DTLZ3 测试函数上,G-RVEA 均严格优于 g-占优和 r-占优;严格优于或弱优于双极偏好占优和 MOEA/D-PRE;在 DTLZ4 测试函数上,除 4 维测试函数以外,G-RVEA 严格优于 g-占优和 r-占优;除 4、6 维测试函数以外,弱优于双极偏好占优;除 10 维目标测试函数以外,弱优于 MOEA/D-PRE. 这表明在 DTLZ 系列测试函数上,G-RVEA 算法收敛性普遍优于其他 4 种偏好算法.

DTLZ2 和 DTLZ4 测试函数是简单连续的单模态测试问题,且初始种群在目标空间中距离真实 Pareto 前沿较近,种群在算法搜索初期就能快速收敛到 Pareto 前沿表面,APD 精英选择策略提升算法收敛性效果并不明显. 在 DTLZ2 和 DTLZ4 测试函数上与 G-RVEA 相比时,g-占优、r-占优、双极偏好占优和 MOEA/D-PRE 的  $I(B,A)$  的值较小,说明 g-占优、r-占优、双极偏好占优和 MOEA/D-PRE 在 DTLZ1 和 DTLZ3 测试函数上的算法收敛性较好,但均劣于 G-RVEA.

DTLZ1 和 DTLZ3 测试函数是复杂多模态测试

问题,存在多个局部最优解,同时其初始种群在目标空间中远离真实 Pareto 前沿,种群收敛到 Pareto 前沿所需迭代次数较多,APD 精英选择策略算法搜索过程中加快种群收敛速度和避免种群退化的效果得以体现. 在 DTLZ1 和 DTLZ3 测试函数上与 G-RVEA 相比时,g-占优、r-占优、双极偏好占优的  $I(B,A)$  值较大,说明 g-占优、r-占优和双极偏好占优在 DTLZ1 和 DTLZ3 测试函数上的算法收敛性较差,基于 Pareto 支配排序的偏好算法难以收敛到 Pareto 前沿上的 ROI 区域;而 MOEA/D-PRE 的  $I(B,A)$  值依然很小,且与  $I(A,B)$  值很接近,说明 MOEA/D-PRE 在 DTLZ1 和 DTLZ3 测试函数上的仍然能维持较好的收敛性,基于分解的偏好算法在复杂高维目标优化问题中能有效降低算法复杂度,将多目标问题转化为多个单目标问题,从而保证算法收敛性;G-RVEA 的  $I(A,B)$  都很小且均优于其他四种偏好算法,表明 G-RVEA 算法在 DTLZ1 和 DTLZ3 等复杂测试函数上所求解集的收敛性最优,APD 精英选择策略通过在各子种群引入 Pareto 非支配你排序,避免种群退化,进一步加快种群的收敛速度.

从纵向上看,我们还可以得出以下结论:(1)随着目标维数的增加, $\epsilon$  指标的  $I(A,B)$  值和  $I(B,A)$  值逐渐增大,说明解的选择压力随着目标维数的增大而增大,导致 g-占优、r-占优、双极偏好占优、MOEA/D-PRE 和 G-RVEA 的算法收敛性随

着目标维数的增加而衰减；(2) 随着目标维数的增加， $I(A, B)$  值的变化幅度很小，说明 APD 精英选择策略能有效缓解高维目标中解的选择压力，使得在 G-RVEA 算法在高维目标优化问题中依然能保持较高的算法收敛性；(3) 随着目标维数的增加，G-RVEA 与 g-占优、r-占优、双极偏好占优的  $I(A, B)$  与  $I(B, A)$  值之间的差值也逐渐增大，而 G-RVEA 与 MOEA/D-PRE 的  $I(A, B)$  与  $I(B, A)$  值相差不大，说明随着目标维数的增加，基于分解的偏好算法在解集收敛性方面的优势更为明显，优于基于 Pareto 支配的偏好算法。

#### 5.4.2 解集均匀性分析

由于在参考点相同的情况下，如果偏好算法所得解集能有效集中到偏好区域，那么其多样性 (HV 指标或  $\Delta$  指标) 反而差。因此，测试 5 种算法所得解集的多样性来衡量算法性能是不合理的。本文采用 SP 指标衡量 5 种对比算法所得解集在目标空间分布中的均匀性，并以此衡量偏好算法在解方向性上的性能优劣。从表 4 所示的 SP 指标测试结果来看，我们可以得出以下结论：

在 3~10 维 DTLZ1-4 测试函数上，G-RVEA 和 MOEA/D-PRE 的 SP 指标值的均值和方差显著优于 g-占优、r-占优和双极偏好占优，说明 G-RVEA 和 MOEA/D-PRE 分别采用种群划分和聚合函数的分解策略，使得个体沿着权重向量方向均匀分布，有效保证种群在高维目标空间分布的高均匀性；g-占优和 r-占优均采用拥挤距离算子维护种群多样性，但是不能保证解集在目标空间中分布的均匀性；双极偏好占优采用  $\delta$  控制策略进一步控制解集的均匀性，因此其 SP 指标略优于 g-占优和 r-占优；

6 维和 8 维 DTLZ4 测试函数外，G-RVEA 的 SP 指标值均略优于 MOEA/D-PRE。这表明在算法搜索后期，APD 精英选择策略优先挑选与各偏好向量夹角更小的个体，迫使个体沿着靠近偏好向量的方向进化，保证 G-RVEA 算法所求解集的高均匀性；而 MOEA/D-PRE 采用邻域更新策略，某一邻域内一旦出现高性能解，则该邻域的所有劣解都将被该解替换，导致种群多样性有所下降，使得 MOEA/D-PRE 算法所求解集均匀性略低于 G-RVEA。

从纵向上看，SP 指标和  $\epsilon$  指标有许多相似之处，具体表现为：随着目标维数的增加，SP 指标的值逐渐增大，说明 g-占优、r-占优、双极偏好占优、MOEA/D-PRE 和 G-RVEA 算法所得解集的均匀性也会随着目标维数的增加而衰减。因为随着

目标维数的增加，逼近偏好区域所需的种群规模呈指数级增长，使得算法所得解集难以覆盖整个偏好区域。

综合表 3 和表 4 的统计结果可以看出，随着目标维数的增加，种群中非支配个体的比例迅速上升，传统 Pareto 支配关系对优秀解的选择压力衰减严重，使得基于 Pareto 占优关系的偏好算法在高维目标测试问题上的整体性能急剧下降；而 G-RVEA 和 MOEA/D-PRE 通过对目标空间进行分解，将复杂的多目标问题转化为多个的单目标问题，同时引入决策者的偏好信息，降低算法在高维目标优化问题中的计算复杂度，确保随着目标维数的增加，算法对种群依然能保持较高的选择压力，加快种群逼近 Pareto 前沿的收敛速度。而 G-RVEA 通过偏好向量生成策略和 APD 精英选择策略，将决策者的偏好融入分解多目标进化算法中，并消除 APD 机制的种群退化现象，进一步平衡算法在高维目标优化问题中所得解集的收敛性和多样性。

图 13 表示 G-RVEA 算法在 3~10 维 DTLZ1-4 测试问题上所得偏好解集的平行坐标系结果图 (实线表示参考点，折线表示 G-RVEA 所得偏好解集)。平行坐标系<sup>[27]</sup>是 Fleming 等人提出的在二维平面上有效表示出高维目标所得解集均匀性的方法，横坐标表示目标维数，纵坐标表示个体在各维目标上的目标函数值，算法所求偏好解集的每个个体在平面坐标系上表现为一条折线。本文提出的 G-RVEA 算法将决策者提供的参考点作为决策者的偏好信息，根据偏好向量生成策略，参考点在 Pareto 前沿上的映射点附近区域就是决策者感兴趣的 ROI 区域，也就是说 G-RVEA 算法所求解集落在映射点附近，就表示满足决策者所需的偏好信息。

G-RVEA 算法在 3~10 维 DTLZ1-4 测试问题上所得偏好解集能很好地满足决策者的偏好，并展现出良好的均匀性。以 10 维 DTLZ2 测试问题为例，参考点  $G_i = 0.2$  在 10 维 DTLZ2 测试问题 Pareto 前沿上的映射点为  $G'_i = 0.316$ ，而与之对应平行坐标系上的个体集中在映射点附近，且个体在各维度上的目标函数值分布得非常均匀。此外，随着目标维数的增加，G-RVEA 算法在 DTLZ1-4 系列测试问题上所求偏好解集依然能集中在决策者感兴趣的 ROI 区域，且保持较高的均匀分布性。这表明随着目标维数的增加，G-RVEA 的算法性能并没有出现明显的衰减。

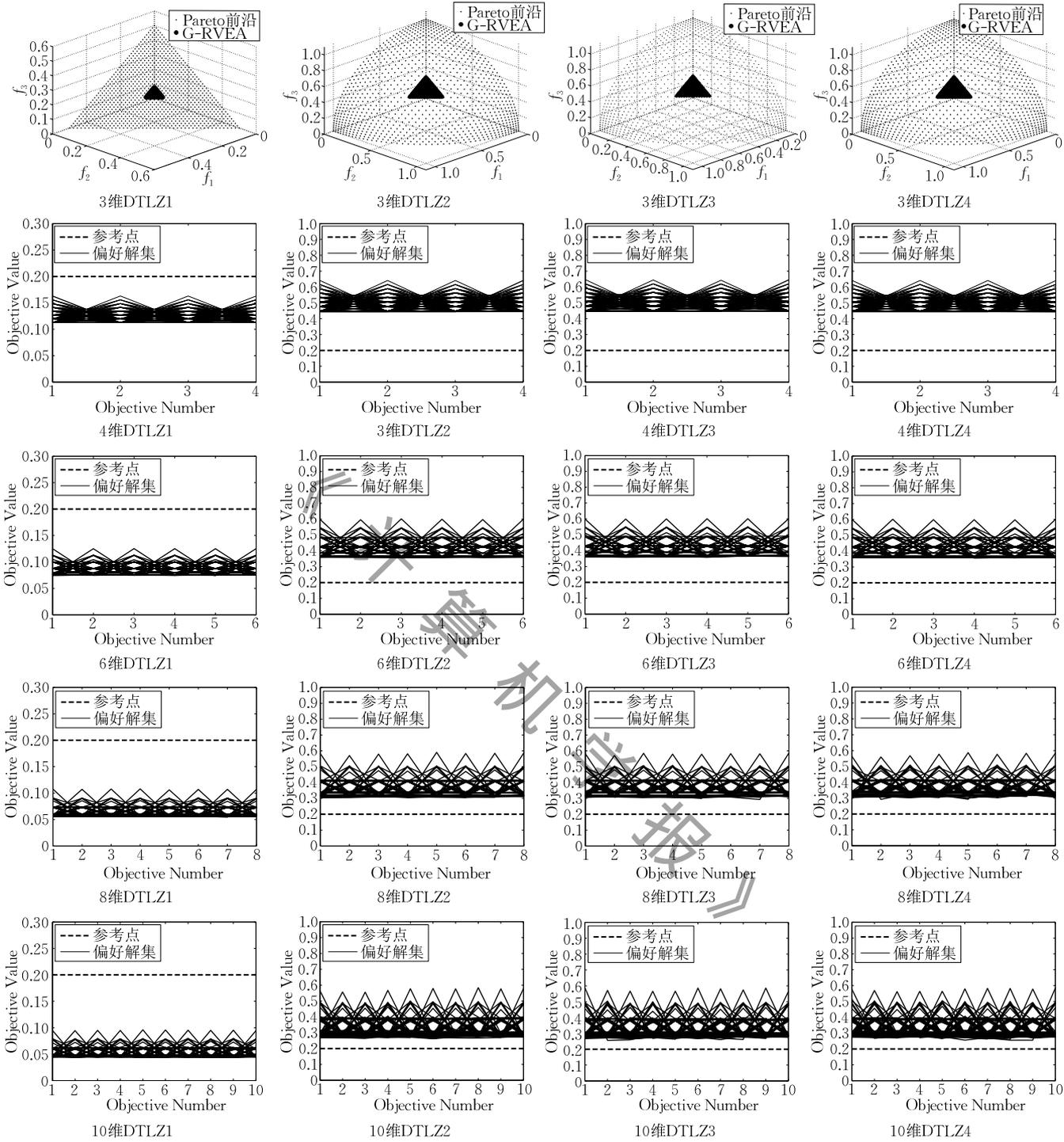


图 13 G-RVEA 在 3~10 维 DTLZ1-4 测试函数上所得偏好解集的平行坐标系结果图

### 5.4.3 算法运行时间对比分析

表 5 是 g-占优、r-占优、双极偏好占优、MOEA/D-PRE 和 G-RVEA 算法在 3~10 维 DTLZ1-4 系列测试函数上的运行时间. 从表 5 的测试结果中可以看出, 在 DTLZ 系列测试函数上, G-RVEA 的算法运行时间远远低于 g-占优、r-占优、双极偏好占优和 MOEA/D-PRE, 这是因为 G-RVEA 通过种群

划分策略, 将目标空间中的个体划分给离其最近的参考向量, 再通过 APD 精英选择策略对各参考向量附近的个体进行筛选, 挑选出 APD 值最小的个体进入下一代, 从而达到降低算法复杂度的目的. g-占优、r-占优和双极偏好占优都是基于传统 Pareto 支配关系的, 随着目标维度的增加, 种群中的非支配解比例越来越高, 算法对优秀解的选择压力下降, 其

中  $r$ -占优对解的筛选方式—加权欧氏距离的算法复杂度随着目标维数的增加呈指数式增长,严重影响算法的运行成本. MOEA/D-PRE 通过将多目标问题转化为多个单一子问题的方式,降低算法复杂度,因此其算法运行时间略差于 G-RVEA.

## 6 总 结

本文将决策者的偏好信息融入到高维目标优化算法中,提出一种基于 APD 精英选择策略的偏好高维目标优化算法 G-RVEA. 该算法首先通过偏好向量生成策略在目标空间中生成一组均匀分布的偏好向量,然后通过种群划分策略将种群中的各个个体划分给与其最近的偏好向量形成子种群,最后通过 APD 精英选择策略筛选子种群中的最优个体,使各个体沿着偏好向量的方向朝着 Pareto 前沿快速收敛. 该算法将决策者偏好信息融入到基于分解的多目标优化算法中,避免参考点位置对算法性能的影响,同时有效分配算法资源,平衡算法收敛性和多样性之间的冲突. 测试证明, G-RVEA 在 3~10 维 DTLZ1-4 标准测试函数上有效地反映决策者的偏好信息,而且在高维目标优化问题上具有良好的算法性能. 该算法为我们研究物流领域的车间调度、工厂选址、路径优化等问题和信息处理等领域中广泛存在的智能计算、网络优化问题的理论研究、数学建模和技术创新提供巨大支持.

## 参 考 文 献

- [1] Brizuela C. A survey on multi-objective evolutionary algorithms for many-objective problems. *Computational Optimization & Applications*, 2014, 58(3): 707-756
- [2] Wang Chao, Jin Chun, Han Jim C. Model and algorithm for multi-objective joint optimization of three-dimensional loading and CVRP. *Control and Decision*, 2016, 31(5): 929-934(in Chinese)  
(王超, 金淳, 韩庆平. 三维装载与 CVRP 联合多目标优化问题的模型及算法. *控制与决策*, 2016, 31(5): 929-934)
- [3] Shen C, Shi Y, Buckham B. Path-following control of an AUV using multi-objective model predictive control// *Proceedings of the 2016 American Control Conference*. Boston, American, 2016: 4507-4512
- [4] Arabali A, Ghofrani M, Etezadi-Amoli M, et al. A multi-objective transmission expansion planning framework in deregulated power systems with wind generation. *IEEE Transactions on Power Systems*, 2014, 29(6): 3003-3011
- [5] Yang Wei, Ban Dong-Song, Guan Dong-Lin, et al. Coalition formation based distributed algorithm for multi-objective cooperative sensing in cognitive radio networks. *Chinese Journal of Computers*, 2012, 35(4): 730-740(in Chinese)  
(杨威, 班冬松, 管东林等. 基于联盟构造博弈的认知无线网络分布式多目标协作感知算法. *计算机学报*, 2012, 35(4): 730-740)
- [6] Hsu C H, Juang C F. Multi-objective continuous-ant-colony-optimized FC for robot wall-following control. *IEEE Computational Intelligence Magazine*, 2013, 8(3): 28-40
- [7] Deb K, Pratap A, Agarwal S, et al. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 2002, 6(2): 182-197
- [8] Zitzler E, Laumanns M, Thiele L. SPEA2: Improving the performance of the strength Pareto evolutionary algorithm// *Proceedings of the Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems*. Athens, Greece, 2001: 95-100
- [9] Zhang Q, Li H. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation*, 2007, 11(6): 712-731
- [10] Kong Wei-Jian, Ding Jin-Liang, Chai Tian-You. Survey on large-dimensional multi-objective evolutionary algorithm. *Control and Decision*, 2010, 25(3): 321-326(in Chinese)  
(孔维健, 丁进良, 柴天佑. 高维多目标进化算法研究综述. *控制与决策*, 2010, 25(3): 321-326)
- [11] Fleming P J, Purshouse R C, Lygoe R J. Many-objective optimization: An engineering design perspective// *Proceedings of the International Conference on Evolutionary Multi-Criterion*. Heidelberg, Berlin, 2005: 14-32
- [12] Deb K, Sundar J. Reference point based multi-objective optimization using evolutionary algorithms// *Proceedings of the 2006 Genetic and Evolutionary Computation Conference*. Kanpur, Indian, 2006: 635-642
- [13] Molina J, Santana L V, Alfredo G. g-dominance: Reference point based dominance for multiobjective metaheuristics. *European Journal of Operational Research*, 2009, 197(2): 685-692
- [14] Said L B, Bechikh S, Dira K. The  $r$ -dominance: A new dominance relation for interactive evolutionary multicriteria decision making. *IEEE Transactions on Evolutionary Computation*, 2010, 14(5): 801-818
- [15] Qiu Fei-Yue, Wu Yu-Shi, Qiu Qi-Cang, Wang Li-Ping. Many-objective evolutionary algorithm based on bipolar preferences dominance. *Journal of Software*, 2013, 24(3): 476-489(in Chinese)  
(邱飞岳, 吴裕市, 邱启仓, 王丽萍. 基于双极偏好占优的高维目标进化算法. *软件学报*, 2013, 24(3): 476-489)
- [16] Zheng Jin-Hua, Yu Guo, Jia Yue. Research on MOEA/D based on user-preference and alternate weight to solve the effect of reference point on multi-objective algorithms. *Acta Electronica Sinica*, 2016, 44(1): 67-76(in Chinese)

(郑金华, 喻果, 贾月. 基于权重迭代的偏好多目标分解算法解决参考点对算法影响的研究. 电子学报, 2016, 44(1): 67-76)

- [17] Wang R, Purshouse R C, Fleming P J. Preference-inspired coevolutionary algorithms for many-objective optimization. *IEEE Transactions on Evolutionary Computation*, 2013, 17(4): 474-494
- [18] Deb K, Jain H. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, Part I: Solving problems with box constraints. *IEEE Transactions on Evolutionary Computation*, 2014, 18(4): 577-601
- [19] Ishibuchi H, Tsukamoto N, Nojima Y. Evolutionary many-objective optimization//*Proceedings of the International Workshop on Genetic and Evolving Systems*. Witten, Germany, 2008: 47-52
- [20] Cheng R, Jin Y, Olhofer M, et al. A reference vector guided evolutionary algorithm for many-objective optimization. *IEEE Transactions on Evolutionary Computation*, 2016, 20(5): 773-791
- [21] Cornell J A. Experiments with mixtures. *Technometrics*, 2011, 44(3): 201-202
- [22] Zitzler E, Deb K, Thiele L. Comparison of multi-objective evolutionary algorithms: Empirical rests evolutionary. *Evolutionary Computation*, 2000, 8(2): 173-195
- [23] Deb K, Thiele L, Laumanns M, Zitzler E. Scalable multi-objective optimization test problems//*Proceedings of the Congress on Evolutionary Computation*. Piscataway, New Jersey, 2002: 825-830
- [24] Van Veldhuizen D A, Lamont G B. On measuring multi-objective evolutionary algorithm performance//*Proceedings of the Congress on Evolutionary Computation*. La Jolla, American, 2000: 204-211
- [25] Zitzler E, Thiele L, Laumanns M. Performance assessment of multiobjective optimizers; An analysis and review. *IEEE Transactions on Evolutionary Computation*, 2003, 7(2): 117-132
- [26] While L, Hingston P, Barone L. A faster algorithm for calculating hypervolume. *IEEE Transactions on Evolutionary Computation*, 2006, 10(1): 29-38
- [27] Inselberg A. The plane with parallel coordinates. *Visual Computer*, 1985, 1(2): 69-91



**WANG Li-Ping**, born in 1964, professor, Ph. D. supervisor. Her main research interests include information processing, decision optimization.

**ZHANG Ming-Lei**, born in 1992, M. S. candidate. His main research interests include intelligent evolutionary algorithm, decision optimization.

**QIU Fei-Yue**, born in 1965, Ph. D., professor. His main research interests include intelligent control, optimization theory and method.

**JIANG Bo**, born in 1985, Ph. D., associate professor. His main research interests focus on intelligent optimization and control.

## Background

This research is supported by the National Natural Science Foundation of China under Grant Nos. 61472366, 61379077, 61503340, and the Zhejiang Provincial Natural Science Foundation of China under Grant Nos. LY17F020022, LQ16F030008.

In this paper, the preference information of the decision makers into multi-objective algorithm based on decomposition, and propose a many-objective optimization algorithm with preference based on the angle penalty distance elite selection strategy, to avoid effect of reference point position on the performance of the algorithm, efficient allocation of resource

management algorithms, and balanced the conflict between convergence and diversity of the algorithm.

Our project aims to provide better performance in many-objective optimization algorithm. Our group has been working on the preferences-based optimization using the bipolar preferences dominance model, the cooperative preferences mechanism and the large scale variable decomposition. New algorithms have been applied into practical work and many high-quality research papers were published in key journals such as *Journal of Software* and *Chinese Journal of Computers*.