

# 改进聚类算子的自适应多目标进化算法研究

谢倩文, 何利力

(浙江理工大学信息学院, 浙江杭州 310018)

**摘要:**对多个相互冲突的目标同时优化称作多目标优化问题,为解决多目标问题,多目标进化算法应运而生。在进化算法迭代过程中,算法使用恒定不变的交叉因子和变异因子,这显然不符合种群迭代进化特征,所以需要根据种群初始和种群后期解的收敛情况来定向调整种群进化方向。同时,在采用边界与交叉的聚合算法时, $\theta$ 支配的聚类算法只参考了解到权重向量的垂直距离,权重向量的投影点到理想点的解的直线距离没有考虑,直接影响了解在 Pareto 前沿上的收敛结果。为此,利用改进后的自适应种群生成策略,动态改变交叉概率和变异概率,根据种群当前的迭代情况调整进化方向;通过增加投影点到理想点的距离和惩罚因子计算个体到聚类中心的距离,将个体放入聚类簇中随机选择,有效提高了算法的收敛性。在多目标问题测试集 ZDT 和 DTLZ 上进行实验,结果表明,NSGA-ACM 在解集的收敛性和分布性上效果更好。

**关键词:**多目标优化;聚类算法;自适应

DOI: 10.11907/rjdk.211331

开放科学(资源服务)标识码(OSID):



中图分类号: TP312

文献标识码: A

文章编号: 1672-7800(2022)001-0146-06

## Multi-Object Optimization Algorithm Based on Improved Adaptive Clustering Algorithm

XIE Qian-wen, HE Li-li

(School of Information Science and Technology, Zhejiang Sci-Tech University, Hangzhou 310018, China)

**Abstract:** Simultaneous optimization of multiple conflicting goals is called a multi-objective optimization problem. Multi-objective evolutionary algorithms are developed to solve these multi-objective problems. In the iterative process of evolutionary algorithm, the algorithm uses constant crossover factor and mutation factor, which obviously does not meet the characteristics of the iterative evolution of the population, so it is necessary to adjust the evolution direction of the population according to the convergence of the initial and late population solutions. At the same time, when using the boundary and cross aggregation algorithm, the clustering algorithm dominated by  $\theta$  only refers to the vertical distance of the weight vector, but the linear distance from the projection point of the weight vector to the ideal point is not considered, which directly affects the individual. The convergence problem on the Pareto frontier. Using the improved adaptive population generation strategy to dynamically change the crossover probability and mutation probability, and adjust the evolution direction according to the current iteration of the population; by increasing the distance between the projection point and the ideal point and the penalty factor, calculate the individual to the center of aggregation. The distance of the individual is randomly selected in the cluster, which effectively improves the convergence of the algorithm. By testing on the multi-objective problem test sets ZDT and DTLZ, NSGA-ACM has a better convergence and distribution effect on the solution set.

**Key Words:** multi-objective optimization; clustering algorithm; self-adaptive

## 0 引言

追求多个目标同时最优化是现代社会生活中经常发

生的事,而几个优化目标在一定程度上互相排斥的情况更是屡见不鲜。因此,多目标优化问题(Multi-objective Optimization Problem, MOP)<sup>[1]</sup>研究发展迅速。多目标进化算法<sup>[2]</sup>可解决多目标优化问题,在遗传算法<sup>[3]</sup>框架下,一系列

收稿日期: 2021-03-08

基金项目: 国家重点研发计划项目(2018YFB1700702)

作者简介: 谢倩文(1996-),女,浙江理工大学信息学院硕士研究生,研究方向为智能软件与数据处理;何利力(1966-),男,博士,浙江理工大学信息学院教授、博士生导师,研究方向为数据分析、企业智能。本文通讯作者: 谢倩文。

的进化算法不断发展,如差分进化算法(Differential Evolution, DE)<sup>[4]</sup>通过改进变异向量收敛于最优解;粒子群优化算法<sup>[5]</sup>(Particle Swarm Optimization, PSO)通过学习鸟类捕食的群体协作行为,利用信息共享获取最优解;蚁群优化算法<sup>[6]</sup>(Ant Colony Optimization, ACO)通过蚁群觅食过程中的信息素引导搜索方向,在组合优化领域取得了较好效果。这些进化算法已经广泛应用于日常生活中,如生产过程中的调度任务最优化<sup>[7]</sup>,社会医疗<sup>[8]</sup>、图像处理<sup>[9]</sup>、神经网络优化、物流配送<sup>[10-11]</sup>以及网络安全<sup>[12]</sup>等领域中的任务优化。随着应用的深入,优化进化算法逐渐成为人工智能技术中的重要一环。因此,研究多目标进化算法对可持续发展具有重大意义。单一目标优化算法只存在单个最优解,而多个目标的优化问题需要尽可能地找到一组最优解集,多个目标优化的核心就是如何在有冲突的多个目标中找到这个最优解集。

多目标进化算法主要包括两部分:①原始种群生成子代种群,利用编码操作为每个个体赋予染色体基因,对父代基因进行交叉和变异生成子代个体;②在子代种群中挑选合适的个体组成新父代进入下一次迭代,选择算法的不同会影响最终所得个体的优秀程度。

## 1 研究现状

解决多目标优化问题算法主要是基于支配关系和基于分解关系。基于支配关系的算法主要以 Deb<sup>[13]</sup>提出的带有精英策略的非支配排序算法 NSGA-II 为主导,在所有解中找到非支配解(Non-dominated solutions)作为 Pareto 最优解(Pareto optimal solutions),利用拥挤距离在非支配解中找出一定数量的优秀解集。NSGA-II 在低维多目标优化问题中效果较好,但是存在搜索能力不足的问题。郑金华等<sup>[14]</sup>提出定向搜索优化策略,前期着重收敛性,后期着重多样性,兼顾了两者的平衡。为解决高维空间中拥挤距离不适用问题,NSGA-III 算法<sup>[15]</sup>利用参考点的约束支配关系找到具有更好收敛性和分布性的最优解集。这些方法证明在一些多目标问题上都有较好性能,但是在一定程度上仍然缺乏维护种群多样性能力。

基于分解方法的算法主要以 Zhang 等<sup>[16]</sup>提出的基于分解的多目标进化算法 MOEA/D 为主导。MOEA/D 方法初始化时先生成一组均匀分布的权重向量,将多目标问题分解成子问题,种群由每个子问题找到的最优解组成。MOEA/D 可扩展性高、收敛快、计算复杂度低,在高维空间中处理结果更理想,但是在相对复杂的 Pareto 前沿情况下,解的分布性会被破坏;Li 等<sup>[17]</sup>提出 MOEA/DD,将支配和分解结合剔除最差解,得到收敛良好和分布良好的点集;Yuan 等<sup>[18]</sup>提出  $\theta$ -DEA,在分解的基础上利用  $\theta$  支配在非支配层上选择解来提高 NSGA-III 的收敛性。但随着目标个数的增加,以 MOEA/D 算法为主要框架的进化算法会出现逼近 Pareto 前沿困难、收敛性下降、最优解选择不足等缺点,在非凸 Pa-

reto 前沿上的分布性还需要进一步优化。本文研究的主要方向是在遗传过程中通过自适应种群来定向调整进化方向以提高收敛速度,并且在收敛过程中根据权重向量的聚类算法对种群进行选择,以保护种群的多样性。

在社会生活中,多目标进化算法用来解决各种问题。针对解决多目标柔性车间调度问题,吕琳<sup>[19]</sup>提出了基于均匀设计的分配方法,提高了求解最优解的能力;周冲<sup>[20]</sup>提出基于参考点的多目标演化算法,在求解卫星星座设计问题中平衡了收敛性和多样性;赵雅卿<sup>[21]</sup>提出基于分解的多目标优化算法,在阵列天线方向中得到很好应用。

## 2 改进的多目标进化算法

基于分解的多目标进化算法 MOEA/D 因操作执行简单且在处理许多目标测试函数时表现良好而受到青睐。在 MOEA/D 算法中,有 3 种分解方法可以衡量权重向量的适应度值,分别是加权求和法(Weighted Sum Approach, WSA)、切比雪夫方法(Tchebycheff Approach, TA)和基于惩罚因子的边界交叉方法(Penalty-based Boundary Intersection approach, PBI)。其中,切比雪夫方法和基于惩罚因子的边界交叉方法因为能够有效处理多目标问题而广泛应用。本文算法需要用到基于惩罚因子的边界交叉聚合方法。PBI 方法根据个体到权重向量和理想点的距离,不断收敛找到 Pareto 前沿的最优解。PBI 计算公式如下:

$$\begin{aligned} \text{minimize } g^{pbi}(x|\lambda, z^*) &= d_1 + \theta \cdot d_2 \\ \text{subject to } x &\in \Omega \end{aligned} \quad (1)$$

其中,  $z^* = (z_1^*, z_2^*, \dots, z_m^*)^T$  是理想点(Ideal point),理想点是由每个目标方向上的最优值组成的一个点,对于任意的  $i = 1, \dots, m$ , 有  $z_i^* = \min \{ f_i^*(x), x \in \Omega \}$ ,  $z_i^*$  表示每个目标方向上的最小函数值。在原始目标方向上移动  $z_i^*$ , 使每个目标方向的最小函数值与该目标方向相切。 $\lambda$  表示均匀分布的权重向量,  $d_1$  代表个体  $f(x)$  到理想点的向量投影在权重向量  $\lambda$  上的投影长度,其值越小表示个体  $f(x)$  越逼近 Pareto 前沿,  $d_2$  表示  $f(x)$  到  $\lambda$  方向的垂直距离。 $\theta$  表示一个惩罚因子可以用来平衡解的分布性和收敛性,通常取  $\theta = 5$ 。 $d_1$  和  $d_2$  的计算公式如下:

$$d_1 = \frac{\|(z^* - F(x))^T \cdot \lambda\|}{\|\lambda\|} \quad (2)$$

$$d_2 = \|F(x) - (z^* - d_1 \cdot \frac{\lambda}{\|\lambda\|})\| \quad (3)$$

NSGA-ACM 算法利用 NSGA-II 的快速非支配排序对种群进行等级排序。结合 MOEA/D 的 PBI 方法,加入惩罚因子  $\theta$  计算  $g^{pbi}$  值,根据  $g^{pbi}$  选出剩余的个体。种群在迭代过程中,利用交叉变异生成子代个体,父子代合并之后根据环境选择形成新种群;若未达到最大迭代次数则继续进行下一次迭代,若达到次数则算法结束,输出最优解。算法如下:

算法 1: NSGA-ACM: Framework of the NSGA-ACM procedure

输入:种群迭代大小,优化问题 MOP

输出:Pareto非支配解

初始化:初始化参照点  $\{\lambda^1, \dots, \lambda^N\}$ , 初始化种群  $N$ , 得到初始理想点  $z^*$

while  $t < \max \text{Gendo}$  (迭代次数未达到)

$\text{Individual} = \text{Recombination} + \text{Mutation}$  (交叉变异进化生成新个体)

$R_t = N \cup \text{Individual}$  (合并子代种群和父代种群)

$P_t = \text{EnvironmentSelection}(R_t)$  (进行环境选择)

$t = t + 1$

end while

return  $P_t$

### 2.1 初始化

根据 Indraneel 等<sup>[22]</sup>提出的 Systematic Approach 产生一组均匀的权重向量  $\{\lambda^1, \dots, \lambda^N\}$  ( $N$  是权重向量的个数, 也即种群大小), 即结构化的参考点。权重向量的个数与优化目标个数  $m$  和分割数  $H$  有关,  $H$  表示在每个目标方向上的分割数, 计算如下:

$$\text{baseSize} = C_{H+m-1}^{m-1} \quad (4)$$

每个参考点满足式(5)要求, 即每个参考点的各分量之和为1。从几何角度看所有参考点都落在一个平面上, 即参考点是基于单位超平面产生的, 且总的个数为  $N$ 。

$$\lambda_i^j \in \left\{ \frac{0}{H}, \frac{0}{H}, \dots, \frac{H}{H} \right\}, \sum_{i=1}^m \lambda_i^j = 1 \quad (5)$$

因此, 初始化  $\{\lambda^1, \dots, \lambda^N\}$  时,  $H$  的选择也需要根据公式来确定。一般来说, 合适的  $H$  值可以使生成的参考点数量等于种群大小  $N$  或者略小于  $N$ 。当  $H = m$  时, 只会在超平面的中心  $(\frac{1}{m}, \frac{1}{m}, \dots, \frac{1}{m})^T$  多出一个参考点, 超平面的其余区域是没有参考点的。然而, 当  $m$  相对较大时, 如  $m = 5$ , 在  $H \geq m$  情形下该算法会产生大量的参考点, 这导致种群的数量异常庞大, 从而出现进化缓慢、求解效率低下问题。所以, 当  $H > 5$  时就需要采用两层权向量生成的方法, 即边界层和内部层。内部层产生的权向量不能直接应用, 需要收缩计算与边界层的权向量在同一个平面上, 最终形成参考点集  $W$ 。

假设边界层和内部层的分割数分别是  $H_1$  和  $H_2$ , 那么最终的参考点个数计算如下:

$$N = N_1 + N_2 = C_{H_1+m-1}^{m-1} + C_{H_2+m-1}^{m-1} \quad (6)$$

通过上述公式计算了边界层和内部层的参考点位置以及数量之后, 内部层参考点需要进行坐标变换。假定原内部层参考点为  $i^k = (i_1^k, i_2^k, \dots, i_m^k)$ ,  $k \in \{1, 2, \dots, H_2\}$ , 其在第  $j$  维的目标函数值进行如式(7)的变换:

$$i_j^k = \frac{1-\tau}{m} + \tau \times i_j^k, j \in \{1, 2, \dots, m\} \quad (7)$$

式中,  $\tau$  是收缩因子, 通常取  $\tau = 0.5$ 。最后, 边界层和内部层的参照点一起合并成最终的参考点集  $Z$ 。当边界层分段数  $p_1 = 2$ 、内部层分段数  $p_2 = 1$  时, 三维目标空间中双层参考点生成情况如图1所示。

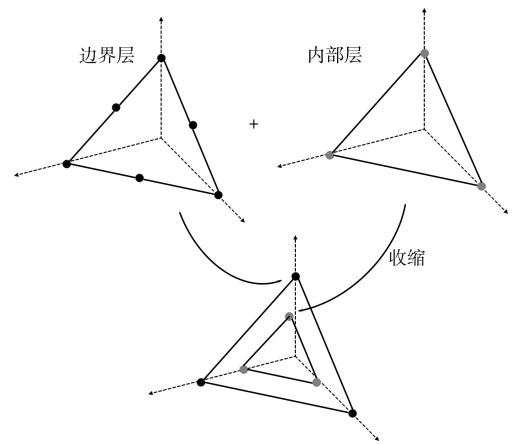


Fig. 1 Two-layer weight vector generation method

图1 两层权向量生成法

### 2.2 新种群生成

对于进化算法来说, 生成子代种群是很重要的环节, 其主要依靠遗传算法的选择操作和重组操作, 而重组操作又分为交叉和变异。选择操作指从父代种群中选择个体生成子代, 而重组操作则是在选择过程中根据交叉因子和变异因子定向改变父代个体, 使得生成的子代个体向最优解收敛。从父代种群中随机选择两个个体  $p_1$  和  $p_2$ , 将  $p_1$  和  $p_2$  进行编码, 对编码基因进行交叉和变异之后生成新的子代个体  $c$ 。在 NSGA-II 算法中选择固定的交叉因子  $P_c$  和变异因子  $P_m$ , 而这两个参数大小直接影响到种群迭代过程中生成的子代个体优秀程度。在种群迭代前期, 小概率的交叉操作可以保证种群的多样性, 交叉因子  $P_c$  需要小一点; 当种群中优秀个体比例较高个体差异不明显之后, 交叉因子  $P_c$  需要大一点。而变异操作能够避免算法陷入局部最优, 种群初期提高变异概率可以加快算法的搜索速度, 种群后期需要保证新种群的优秀个体不被淘汰, 所以选择较小的变异概率。

为了在迭代过程中提高收敛速度并且丰富解的多样性, 本文提出了自适应交叉变异策略以保证每一次的交叉和变异都是根据当前种群解的情况所得到, 计算公式如式(8)和式(9)所示。

$$P_c = \varepsilon - \frac{2e^{-\frac{i}{gen}}}{1 + e^{-\frac{i}{gen}}} \times P_c \quad (8)$$

$$P_m = \frac{2e^{-\frac{i}{gen}}}{1 + e^{-\frac{i}{gen}}} \times P_m \quad (9)$$

其中,  $P_c$  表示算法固定的交叉因子,  $P_m$  表示算法初始的变异因子,  $i$  表示当前的第  $i$  次迭代,  $gen$  表示算法设置的种群最大迭代次数, 根据实验情况设置为 1.5 时效果最好。本文采用模拟二进制交叉 (SBX) 和多项式变异 (PM) 的方法来产生子代种群。通过模拟单点二进制交叉方法, 两个父代个体  $X^1(x_1^1, \dots, x_n^1)$  和  $X^2(x_2^1, \dots, x_n^2)$  通过使用 SBX 算子产生两个后代个体  $C^1(c_1^1, \dots, c_n^1)$  和  $C^2(c_1^2, \dots, c_n^2)$ 。

$$\begin{aligned} c_i^1 &= 0.5 \times [(1 + \beta) \cdot x_i^1 + (1 - \beta) \cdot x_i^2] \\ c_i^2 &= 0.5 \times [(1 - \beta) \cdot x_i^1 + (1 + \beta) \cdot x_i^2] \end{aligned} \quad (10)$$

其中,  $\beta$  是由分布因子  $\eta$  按照公式(11)动态随机决定的。 $\eta$  是一个自定义的参数,  $\eta$  值越大, 产生的后代个体逼近父代个体的概率越大, 建议  $\eta=1$ 。

$$\beta = \begin{cases} (\text{rand} \times 2)^{1/(\eta+1)}, \text{rand} \leq 0.5 \\ (1/(2 - \text{rand} \times 2))^{1/(\eta+1)}, \text{其他} \end{cases} \quad (11)$$

多项式变异操作使个体基因改变, 提高了算法的搜索能力, 计算公式如式(12)和式(13)所示。其中,  $i$  表示决策向量  $x$  的第  $i$  个分量,  $a_i$  和  $b_i$  分别表示第  $i$  个分量的上界和下界,  $p_m$  表示变异概率, 只有达到变异概率才会发生变异。

$$x_i = \begin{cases} x_i + \Delta i \times (b_i - a_i), & \text{with probability } p_m \\ x_i, & \text{with probability } 1 - p_m \end{cases} \quad (12)$$

$\Delta i$  的计算公式如式(13)所示。其中, 随机数  $u$  是  $[0, 1]$  的随机数,  $\eta$  是一个自定义的分布参数, 为任意非负实数。遗传算法通过交叉和变异这对相互配合又相互竞争的操作使其具备兼顾全局和局部的均衡搜索能力, 使算法维持群体多样性。

$$\Delta_j = \begin{cases} (2u_j)^{\frac{1}{\eta+1}} - 1, u_j < 0.5 \\ 1 - (2(1 - u_j))^{\frac{1}{\eta+1}}, \text{其他} \end{cases} \quad (13)$$

### 2.3 环境选择

环境选择的目的是选择出当前种群中符合最优条件的个体进行下一次迭代, 环境选择算法如下:

算法 2: *Environmental Selection*( $R, t$ )

输入: 解集  $R_t$  和迭代次数  $t$

输出: 下一代种群  $P_{t+1}$

$(F_1, F_2, \dots) = \text{Fast-non-dominated sorting}(R_t)$  //非支配排序

$P_{t+1} = \emptyset, i = 1$

while  $|P_{t+1}| + |F_i| \leq N$  do //关键层之前的非支配解全部放入新种群中

$P_{t+1} = P_{t+1} \cup F_i$  and  $i = i + 1$

End while

$F_i = F_i$

if  $|P_{t+1}| \leq N$  then (在关键层中选最后  $K$  个)

$C_1, \dots, C_N = \text{Clustering}(F_i, A, D_{\text{intercept}}), i \in \{1, \dots, N\}$

将关键层中的个体根据  $g^{\text{pipbi}}$  的值排序,  $G_j =$

$\bigcup_{i=1}^N C_i, j \in \{1, \dots, N\};$

( $C_i$  表示第  $i$  个聚集中心的第  $j$  个个体,

$G_j$  表示所有聚集中心的第  $j$  个个体的集合)

$r = 1$

While  $|P_{t+1}| + |G_r| \leq N$  do

$P_{t+1} = P_{t+1} \cup G_r$  (从距离参照点最近的个体开始依次放入新种群中)

$r = r + 1$

end while (直到新种群数量为  $N$  时为止)

从  $G_r$  中随机选择  $N - |P_{t+1}|$  个体

end if

return  $P_{t+1}$

初始化时, 种群  $P_0$  和第一代子代种群  $Q_0$  混合形成新种群  $R$ , 迭代选择从种群  $R$  开始。本文根据 NSGA-II 中的快速非支配排序算法将种群  $R$  按照非支配等级进行排序。首先找出当前种群  $R$  中的非支配解集, 记为第一非支配层  $F_1$ , 该非支配解集中的解互不支配, 但是能支配不在这个解集中的其他所有解。将  $F_1$  的所有解赋予非支配序号  $i_{\text{rank}} = 1$  (其中  $i_{\text{rank}}$  是个体  $i$  的非支配序值), 除去第一非支配层  $F_1$  之后, 在剩下的种群个体中找到新的非支配解集, 记为第二非支配层  $F_2$ ,  $F_2$  中解的非支配序号在上一支配层的基础上增加为  $i_{\text{rank}} = 2$ , 序号越低, 解的支配优先级越高。按照上述步骤进行下去, 直到整个种群分为不同支配级别的解集, 同一层内的个体具有相同的非支配序号  $i_{\text{rank}}$ 。将种群划分成不同等级之后, 先将优先级最高的等级层中的个体放入新种群  $S$ , 若放入新种群之后数量仍然小于  $N$ , 则继续加入下一个优先级等级层的个体, 直到  $S$  的数量大于或等于  $N$ 。若加入  $F_i$  层时新种群数量大于  $N$ , 则需要在  $F_i$  层中选择部分合适的个体放入  $S$  直到数量等于  $N$ , 此时的  $F_i$  层称为关键层。

NSGA-II 算法在关键层中选择合适的剩余个体是通过计算关键层中个体的拥挤距离来实现的, 按照从大到小依次将个体放入新种群  $S$  中。但是拥挤距离在超过 3 维的多目标进化算法中选择效果不太理想, 因此本文采用基于惩罚的边界交叉聚合方法来选择种群个体。对于每一个权重向量, 计算该非支配层上所有个体到权重向量的  $g^{\text{pbi}}$  值, 将个体分配到具有最小  $g^{\text{pbi}}$  值的聚簇  $C_j$  中。依次将各个聚簇中心的个体放入新种群中, 直到新种群的数量为  $N$  时停止。

在生成新子代过程中, 基于固定交叉因子和变异因子的多目标进化算法 NSGA-II 不能根据种群的迭代情况动态调整参数, 导致种群前期和后期在拥有不同占比优秀解的情况下算法搜索能力下降和收敛速度减缓。同时, 在关键层使用拥挤距离来选择个体并不适合高维度优化问题, 这直接影响了在 Pareto 前沿上的分布和收敛效果。本文利用改进后的自适应种群生成策略, 动态地改变交叉概率和变异概率, 根据种群当前的迭代情况调整进化方向, 同时通过关联个体和权重向量改变个体与权重向量聚合的方法, 由此选择出最后的剩余个体, 形成新一代种群, 有效提高算法的收敛性。通过多目标问题测试集 ZDT 和 DTLZ 上进行测试, NSGA-ACM 在解集的收敛性和分布性上效果更好。

## 3 仿真实验及分析

### 3.1 算法性能评价指标

参数设置见表 1。

分布性(Distribution): 解是否完全落在整个 Pareto 前沿上, 两个解之间的分布是否与 Pareto 前沿的形状相适应。

Table 1 Parameter settings

表1 参数设置

参数	参数值
目标维度	2/5
种群规模	100/300
最大迭代次数	300/500
初始交叉因子	$P_c = 1.0$
初始变异因子	$P_m = 1/N$

收敛性 (Convergence): 算法得到的解与真实的 Pareto 前沿的距离<sup>[23]</sup>显示, 真实解距离 Pareto 前沿越近, 越能找到近似最优解。

逆世代距离 (Inverted Generational Distance, IGD): 表示真实 Pareto 前沿中每个解到近似解中所有解的欧氏距离, 从而对算法的收敛性和分布性进行评估<sup>[24]</sup>。IGD 越小表示得到的解的综合收敛效果和分布效果越好。如果 IGD 值为 0, 则说明得到的候选解覆盖了全部真实 Pareto 前沿。

### 3.2 算法性能测试对比

#### 3.2.1 测试实验

本文主要在二维优化问题和五维优化问题上进行测试。在二维优化问题上, NSGA-II 和 MOEA/DD 的交叉因子  $P_c$  和变异因子  $P_m$  分别设置为 0.9 和  $1/N$ , 而改进后的 NSGA-ACM 的交叉和变异因子是根据种群的迭代次数自适应变化调整的, 但  $P_c$  和  $P_m$  的初始值为 0.9 和  $1/N$ 。本实验选择在 ZDT2 测试问题上进行对比测试。ZDT2 是非凸问题, 能够较全面地测试 NSGA-ACM 在 Pareto 前沿的分布性和收敛性。在五维优化问题中, 交叉因子  $P_c$  和变异因子  $P_m$  的设置与上述二维优化问题设置相同, 对比实验选择在 DTLZ 问题上进行, 设置最大迭代次数  $maxGen=300$ , 种群大小  $N=300$ 。图 2 和图 3 分别是算法在 ZDT2 问题和 DTLZ1 问题上的测试结果。

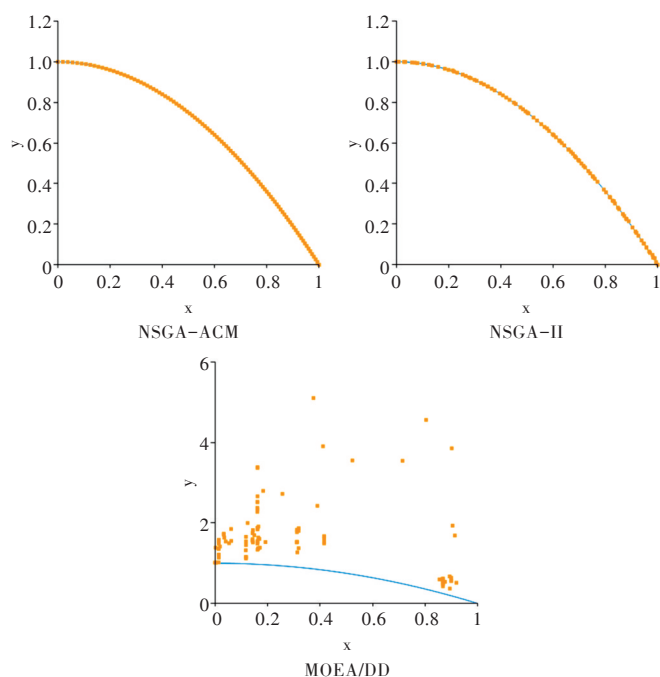


Fig. 2 Test result of ZDT2

图2 ZDT2测试结果

参数初始化设置多目标问题为 DTLZ1 问题, 种群的最大迭代次数  $gen=300$ 。

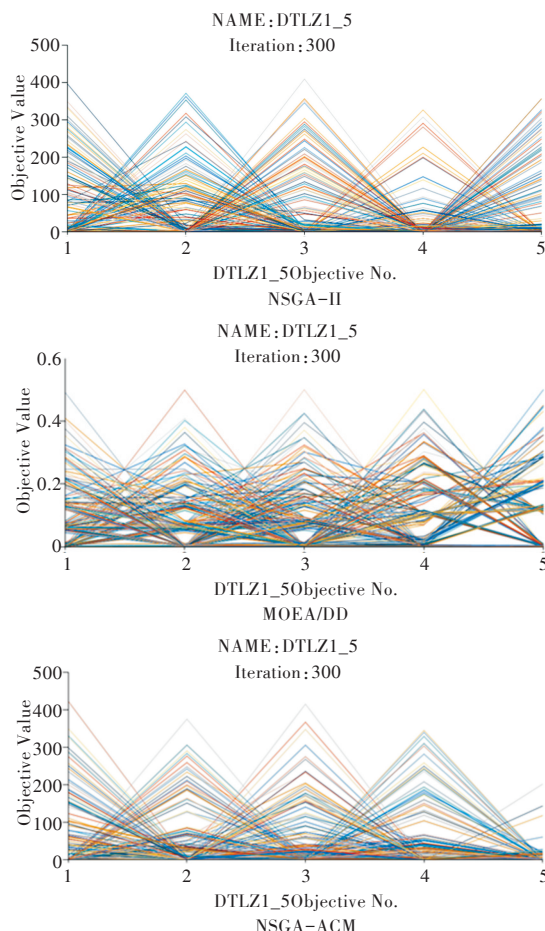


Fig. 3 Test result of DTLZ1

图3 DTLZ1测试结果

#### 3.2.2 测试结果分析

如图 2 所示, NSGA-ACM 算法在二维优化问题中找到的解基本上都处在真实 POF 上, 并且解在中心区域和靠近坐标轴区域仍然保持着均匀分布; NSGA-II 算法中虽然大部分解都能落在 POF 上, 但最终的分布效果不是很理想, 有待于进一步优化; MOEA/DD 算法的分布性和收敛性都不如 NSGA-ACM 算法, 部分解无法收敛到 POF。表 2 为算法在测试函数上的 IGD 值的平均值和标准差。从表中数据可知, NSGA-ACM 在测试函数上的 IGD 值要略低于其他算法, 并且在重复实验中算法的 IGD 值较为稳定, 上下浮动不大。在 DTLZ 测试函数上, NSGA-ACM 收敛性比 NSGA-II 更好, 分布性比 MOEA/DD 更广。综上所述, NSGA-ACM 在二维以及五维优化问题上的实现效果优于 MOEA/DD 以及 NSGA-II。

Table 2 IGD mean and standard deviation of each algorithm of different test functions

表2 在不同测试函数上各算法的 IGD 的平均值和标准差

Prob.	Type	MOEA/DD	NSGA-II	NSGA-ACM
ZDT1	mean	3.0184E-02	3.8247E-03	3.8178E-03
	standard	3.8653E-03	1.3675E-05	1.2874E-05
DZTL1	mean	1.6486E-02	1.6233E-02	1.0895E-02
	standard	1.2056E-05	4.5745E-05	2.8512E-06

## 4 结语

NSGA-ACM 算法采用自适应种群的交叉因子和变异因子定向调整种群迭代方向,在种群初期增强种群的多样性,在迭代后期又能维护种群的多样性,此种改进方式使得算法能够搜索到更加符合真实 Pareto 前沿的最优解。在 PBI 方法的基础上通过个体与参考点关联策略,将个体放在参考点的聚簇中,找出剩余最优解,提高了算法的收敛性。在二维测试函数上进行测试实验,对比分析显示该算法不仅能够收敛于 Pareto 前沿,还能保证完全覆盖 Pareto 前沿以及在 Pareto 前沿上均匀分布。从各项实验数据可知,NSGA-ACM 算法在二维和五维空间中,与 NSGA-II、MOEA/DD、NSGA-II 这 3 个算法对比都能展示出更优越的性能,在均匀分布性以及收敛性方面都得到了改善。

### 参考文献:

- [1] MIETTINEN K. Nonlinear multiobjective optimization [M]. Berlin: Springer Science & Business Media, 2012.
- [2] GONG M G, JIAO L C, YANG D D, et al. Research on evolutionary multi-objective optimization algorithm [J]. Journal of Software, 2009, 20(2): 271-289.  
公茂果,焦李成,杨咚咚,等. 进化多目标优化算法研究 [J]. 软件学报, 2009, 20(2): 271-289.
- [3] HOLLAND J H. Genetic algorithms and adaptation [M]. Berlin: Springer, 1984: 317-333.
- [4] STORN R, PRICE K. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces [J]. Journal of Global Optimization, 1997, 11(4): 341-359.
- [5] EBERHART R, KENNEDY J. A new optimizer using particle swarm theory [C]//Proceedings of the Sixth International Symposium on Micro Machine and Human Science, 1995: 39-43.
- [6] COLORNI A, DORIGO M, MANIEZZO V. Distributed optimization by ant colonies [C]//Proceedings of the First European Conference on Artificial Life, 1992: 134-142.
- [7] YI T, LEI D M. A survey of researches on artificial bee colony for production scheduling [J]. Journal of Hebei University of Technology, 2020, 49(4): 24-32.  
易天,雷德明. 基于人工蜂群算法的生产调度研究进展 [J]. 河北工业大学学报, 2020, 49(4): 24-32.
- [8] LIN S F, LIN F, CHEN H S, et al. A MOEA/D-based multi-objective optimization algorithm for remote medical [J]. Neurocomputing, 2017, 220: 5-16.
- [9] LIU K, GONG D W. Multi-objective optimization model and its evolution-based solution for gesture segmentation problems [J]. Control and Decision, 2017, 32(1): 100-104.  
刘可, 巩敦卫. 手势分割问题的多目标优化模型及其进化求解方法 [J]. 控制与决策, 2017, 32(1): 100-104.
- [10] TAN K C, CHEW Y H, LEE L H. A hybrid multi-objective evolutionary algorithm for solving truck and trailer vehicle routing problems [C]//Canberra: The 2003 Congress on Evolutionary Computation, 2003.
- [11] ABEL G N, JOHN A B. An improved multi-objective evolutionary algorithm for the vehicle routing problem with time windows [J]. Computers & Operations Research, 2011, 38(1): 287-300.
- [12] CHUGH T, SINDHYA K, HAKANEN J, et al. A survey on handling computationally expensive multiobjective optimization problems with evolutionary algorithms [J]. Soft Computing, 2019, 23(9): 3137-3166.
- [13] DEB K. A Fast and elitist multiobjective genetic algorithm: NSGA-II [J]. IEEE Transactions on Evolutionary Computation, 2002, 6(2): 182-197.
- [14] ZHENG J H, DONG N J, RUAN G, et al. High-dimensional multi-objective optimization strategy based on decision space oriented search [J]. Journal of Software, 2019, 30(9): 2686-2704.  
郑金华,董南江,阮干,等. 决策空间定向搜索的高维多目标优化策略 [J]. 软件学报, 2019, 30(9): 2686-2704.
- [15] 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.
- [16] 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.
- [17] LI K, KALYANMOY DEB, ZHANG Q F, et al. An evolutionary many-objective optimization algorithm based on dominance and decomposition [J]. IEEE Transactions on Evolutionary Computation, 2015, 19(5): 694-716.
- [18] 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, 2016, 20(1): 16-37.
- [19] LYU L. Research and application of improved adaptive multi-objective genetic algorithm [D]. Tianjin: Tianjin University of Technology and Education, 2020.  
吕琳. 基于改进的自适应多目标遗传算法的研究及其应用 [D]. 天津: 天津职业技术师范大学, 2020.
- [20] ZHOU C. Research of reference points based many-objective evolutionary algorithm and application to satellite constellation design problem [D]. Wuhan: China University of Geosciences, 2018.  
周冲. 基于参考点的高维多目标演化算法研究及其在卫星星座设计中的应用 [D]. 武汉: 中国地质大学, 2018.
- [21] ZHAO Y Q. Improved multi-objective optimization algorithm based on decomposition and its application in array antenna pattern [D]. Changchun: Jilin University, 2020.  
赵雅卿. 改进的基于分解的多目标优化算法及其 [D]. 长春: 吉林大学, 2020.
- [22] INDRANEEL D, DENNIS J. Normal-boundary intersection: a new method for generating the pareto surface in nonlinear multicriteria optimization problems [J]. Siam Journal on Optimization, 1998, 8(3): 631-657.
- [23] LI M Q, ZHENG J H. An indicator for assessing the spread of solutions in multi-objective evolutionary algorithm [J]. Chinese Journal of Computers, 2011, 34(4): 647-664.  
李密青, 郑金华. 一种多目标进化算法解集分布广度评价方法 [J]. 计算机学报, 2011, 34(4): 647-664.
- [24] ZHANG Y J. Research on many-objective evolutionary algorithms and their applications [D]. Harbin: Harbin Engineer University, 2015.  
张永建. 高维多目标进化算法研究与应用 [D]. 哈尔滨: 哈尔滨工程大学, 2015.