



控制与决策

CONTROL AND DECISION



基于聚类引导和目标值和高维多目标进化算法

王旭健, 张峰干, 姚敏立

引用本文:

王旭健,张峰干,姚敏立. 基于聚类引导和目标值和高维多目标进化算法[J]. 控制与决策, 2024, 39(10): 3190–3198.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2023.0596>

您可能感兴趣的其他文章

Articles you may be interested in

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R2指标和目标空间分解的高维多目标粒子群优化算法

R2 indicator and objective space partition based many-objective particle swarm optimizer

控制与决策. 2021, 36(9): 2085–2094 <https://doi.org/10.13195/j.kzyjc.2020.0113>

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>

基于聚类引导和目标值和高维多目标进化算法

王旭健[†], 张峰干, 姚敏立

(火箭军工程大学 作战保障学院, 西安 710025)

摘要: 基于分解的高维多目标进化算法在处理不规则前沿优化问题时需要调整参考向量, 为避免这一复杂操作, 提出一种基于聚类引导和目标值和高维多目标进化算法. 该算法借助一个储存非支配解并定期更新的精英集, 通过聚类引导当前种群进化, 从而使得当前种群保持较好的多样性. 选择个体时, 根据 Pareto 支配关系以及目标值和衡量个体的收敛性, 基于该收敛性度量方式进行非支配排序和适应值排序, 从而选择收敛性较好的个体. 与7种算法在2套高维多目标优化测试题上进行对比实验, 实验结果表明, 所提出算法能有效解决不同类型的高维多目标优化问题.

关键词: 多目标优化; 高维多目标优化; 分解; 聚类; 目标值和

中图分类号: TP18

文献标志码: A

DOI: 10.13195/j.kzyjc.2023.0596

引用格式: 王旭健, 张峰干, 姚敏立. 基于聚类引导和目标值和高维多目标进化算法 [J]. 控制与决策, 2024, 39(10): 3190-3198.

A many-objective evolutionary algorithm based on clustering and the sum of objectives

WANG Xu-jian[†], ZHANG Feng-gan, YAO Min-li

(College of Combat Support, Rocket Force University of Engineering, Xi'an 710025, China)

Abstract: Decomposition-based many-objective evolutionary algorithms need to adjust reference vectors when solving problems with irregular Pareto fronts. To avoid this complicated operation, this paper proposes a many-objective evolutionary algorithm based on clustering and the sum of objectives (CSEA). This algorithm introduces a periodically updated elitist archive to store non-dominated solutions, which guides the evolving directions of the current population through clustering and maintains the diversity of the current population. When selecting solutions, CSEA evaluates convergence according to Pareto dominance and the sum of objectives, and then select well-converged solutions according to non-dominated sorting and fitness-based sorting. Compared with seven algorithms on two many-objective optimization test suites, CSEA is effective on many-objective optimization problems with various shapes of Pareto fronts.

Keywords: multiobjective optimization; many-objective optimization; decomposition; clustering; sum of objectives

0 引言

高维多目标优化问题 (many-objective optimization problems, MaOPs) 一般指优化目标多于3个的优化问题, 各优化目标相互冲突, 不存在唯一最优解使各目标同时达到最优, 因此, 寻找能平衡多个目标优劣的 Pareto 最优解集, 成为解决此类问题的关键^[1]. 高维多目标进化算法 (many-objective evolutionary algorithms, MaOEAs) 基于种群设计, 一次运行可以得到多个解, 同时不要求优化问题具有可微等数学特性, 成为解决 MaOPs 的有效方法^[2]. 根据

选择策略的不同, MaOEAs 一般可分为3类, 即基于支配关系、基于指标、基于分解的算法^[3-4].

基于支配关系的算法通过修改 Pareto 支配或者定义新的支配关系来改变支配区域大小, 使 Pareto 支配无法区分的个体在新的支配关系下得以区分, 如模糊支配^[5]、网格支配^[6]、强化支配^[7-8]等. 这类算法会引入新参数, 参数调整使算法变得更复杂. 此外, 这类算法容易使种群向 Pareto 前沿面的部分区域收敛.

基于指标的算法利用性能指标在进化过程中选择个体, 如 IGD^[9]、HV^[10-11]、R2^[12-13]等, 这些指标不受

收稿日期: 2023-05-06; 录用日期: 2023-10-01.

责任编辑: 陈家伟.

[†]通讯作者. E-mail: wxj_903@163.com.

优化问题 Pareto 前沿形状的影响,但是在高维目标空间中计算复杂度高,计算时间长。

基于分解的算法利用预先设定的均匀分布的参考向量,将原优化问题分解成一系列子问题,通过同时优化这些子问题实现对原优化问题的求解^[14-16]。在基于分解的算法中,常假设若参考向量均匀分布,则得到的最优解集中的个体也将均匀分布^[17-18]。随着研究深入,学者们发现,当优化问题的 Pareto 前沿形状不规则时,参考向量的均匀分布并不能保证最优解集中个体的均匀分布^[19]。对此,文献^[20]综述了在进化过程中调整参考向量分布的方法,然而,这一方法需要设计调整时机、调整频率、调整策略等复杂因素,这些因素直接影响最终得到的最优解集,使得这类方法无法有效解决复杂 Pareto 前沿形状的优化问题。最近,有文献研究利用机器学习方法调整参考向量,如生长神经网络^[21]、增量学习^[22]等,但这类方法涉及学习和训练过程,使算法复杂度增加,耗费更多计算资源。

针对上述问题,为避免调整参考向量,本文提出一种基于聚类引导和目标值和高维多目标进化算法(CSEA)。文献^[18]指出,在进化过程中,非支配解逐渐逼近优化问题的最优 Pareto 前沿,非支配解的分布可以反映最优 Pareto 前沿的形状。基于此,CSEA 利用一个储存非支配解的精英集(elitist archive, EA),通过聚类引导当前种群进化,同时,通过非支配排序和小生境策略定期更新该精英集,使其具有良好的收敛性和多样性。选择个体时,根据 Pareto 支配和目标值和衡量个体的收敛性,基于此收敛度量方式,利用非支配排序和适应值排序选择收敛性较好的个体。将本文方法与 7 种算法在 2 套测试题上进行对比实验,结果表明,CSEA 可有效解决不同类型的高维多目标优化问题。

1 MaOP 数学模型和相关概念

不失一般性,以最小化为例, MaOP 可用以下数学模型表示:

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

其中: $\mathbf{x} = (x_1, \dots, x_D) \in R^D$ 表示决策空间 Ω 中的决策向量, D 表示决策向量中变量的个数, $\mathbf{F}(\mathbf{x})$ 表示包含 $m (m \geq 4)$ 个优化目标的目标向量。下面给出 MaOP 中的几个基本概念。

1) Pareto 支配。

对于 $\mathbf{x} \in \Omega, \mathbf{y} \in \Omega$, 称 \mathbf{x} Pareto 支配 \mathbf{y} , 记作 $\mathbf{x} \prec \mathbf{y}$, 当且仅当

$$\begin{aligned} \forall i = 1, 2, \dots, m, f_i(\mathbf{x}) &\leq f_i(\mathbf{y}); \\ \exists j = 1, 2, \dots, m, f_j(\mathbf{x}) &< f_j(\mathbf{y}). \end{aligned} \quad (2)$$

2) Pareto 最优解集(optimal Pareto set, PS)。

对于 $\mathbf{x}^* \in \Omega$, 称 \mathbf{x}^* 为 Pareto 最优解, 当且仅当 $\neg \exists \mathbf{x} \in \Omega$, 使 $\mathbf{x} \prec \mathbf{x}^*$ 成立。所有 Pareto 最优解构成的集合称为 Pareto 最优解集。

3) Pareto 最优前沿(optimal Pareto front, PF)。

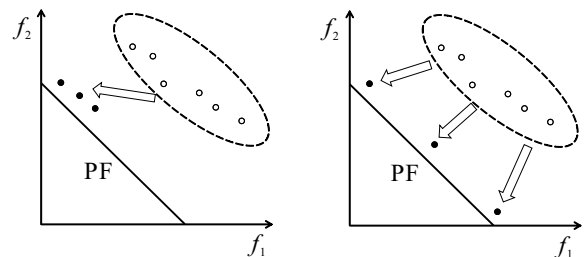
Pareto 最优前沿是 Pareto 最优解集从决策空间向目标空间的映射, 定义为

$$\text{PF} = \{\mathbf{F}(\mathbf{x}^*) = (f_1(\mathbf{x}^*), \dots, f_m(\mathbf{x}^*)) | \mathbf{x}^* \in \text{PS}\}. \quad (3)$$

2 基于聚类引导和目标值和高维多目标进化算法

2.1 算法框架

在 CSEA 中, 用于引导当前种群进化的精英集需要有较好的收敛性和多样性, 否则, 精英集会引导当前种群向错误的方向进化。如图 1 所示, 以 2 目标优化问题为例, 实心点表示精英集中的个体, 空心点表示进化种群中的个体, 箭头表示进化方向。在图 1(a) 中, 精英集中的个体聚集在 PF 的部分区域, 引导进化种群中的个体向聚集的方向进化, 不利于搜索整个目标空间。只有当精英集中的个体具有较好的多样性时, 如图 1(b) 所示, 精英集才能引导当前种群中的个体向正确的方向进化。



(a) 多样性差的精英集 (b) 多样性好的精英集

图 1 精英集的进化引导作用示意

为保证精英集正确引导当前种群进化, CSEA 将整个进化过程分为两个阶段。第 1 阶段为进化初期, 在 CSEA 中定义为进化代数小于 10% 的最大进化代数, 或者精英集规模 $|EA|$ 小于种群规模 N , 此时精英集不具有好的收敛性和多样性, 因此这一阶段的环境选择不使用精英集。除第 1 阶段外的其他时期为第 2 阶段, 此时精英集可以反映 PF 的分布, 因此这一阶段借助精英集进行环境选择。此外, 为保持精英集的收敛性和多样性, 还需对其定期更新。CSEA 的具体流程如算法 1 所示。

算法1 CSEA基本流程.

输入: 种群规模 N , 最大迭代次数 FE_{\max} , 精英集更新频率 f_r ;
 输出: 最终解集 P .

- 1: 随机生成初始种群 P ;
- 2: 对 P 进行非支配排序, 得到层级序号 $FrontNo$;
- 3: 初始化精英集 $EA = P(FrontNo == 1)$;
- 4: while $FE \leq FE_{\max}$ do
- 5: $P' =$ 匹配选择 ($P, FrontNo$);
- 6: $P'' =$ 交叉变异操作 (P');
- 7: if $FE \leq 0.1 \times FE_{\max}$ or $|EA| < N$ then
- 8: $[P, FrontNo] =$ 第1阶段环境选择 ($P \cup P'', N$);
- 9: else
- 10: $[P, FrontNo] =$ 第2阶段环境选择 ($P \cup P'', EA, N$);
- 11: if $\text{mod}(\text{ceil}(FE/N), \text{ceil}(f_r \times FE_{\max}/N)) == 0$ then
- 12: $EA =$ 精英集更新 ($EA \cup P'', N$);
- 13: return P .

首先, 从决策空间中随机生成包含 N 个个体的初始种群 P , 对 P 进行非支配排序^[23] 得到每个个体的非支配层级序号 $FrontNo$, 初始化精英集为非支配解的集合; 其次, 按照下述过程迭代优化直至最大迭代次数: 通过匹配选择生成匹配池, 对匹配池中的个体进行模拟二进制交叉和多项式变异操作, 生成子代种群; 然后, 根据不同阶段进行相应的环境选择; 最后, 对精英集进行定期更新以引导种群进化.

2.2 匹配选择

匹配选择旨在从父代种群中挑选个体组成匹配池. 由算法1第5行知, CSEA的匹配选择根据 $FrontNo$ 完成, 且对应两个进化阶段, $FrontNo$ 具有不同含义. 在第1阶段, $FrontNo = NDFrontNo$, 其中 $NDFrontNo$ 是指非支配排序得到的非支配层级序号; 在第2阶段, $FrontNo = [NDFrontNo, FitFrontNo]$, 其中 $FitFrontNo$ 是指适应值排序得到的适应值层级序号(将在2.3.2节介绍).

匹配选择时, 从父代种群中任选两个个体进行比较: 在进化第1阶段, 选择 $NDFrontNo$ 较小的个体进入匹配池, 若两者 $NDFrontNo$ 相同, 则随机选择一个个体. 在进化第2阶段, 先选择 $NDFrontNo$ 较小的个体, 若两者 $NDFrontNo$ 相同, 则再选择 $FitFrontNo$ 较小的个体; 若两者 $NDFrontNo$ 和 $FitFrontNo$ 都相同, 则随机选择一个个体. 重复上述操作, 直至选择出 N 个个体.

2.3 环境选择**2.3.1 第1阶段环境选择**

在进化的第1阶段, 精英集无法反映PF形状, 不能正确引导种群进化, 因此这一阶段的环境选择不使用精英集, 而依次使用非支配排序和“最大化最小角度”策略^[24] 选择个体. 具体过程如算法2所示.

算法2 第1阶段环境选择.

输入: 混合种群 Q , 种群规模 N ;
 输出: 下一代种群 P , 层级序号 $FrontNo$.

- 1: 对 Q 进行非支配排序, 得到非支配层级序号 $NDFrontNo$;
- 2: $Q = [F_1, \dots, F_l]$ // l 是满足 $|F_1 \cup \dots \cup F_l| \geq N$ 的最小整数;
- 3: $P = [F_1, \dots, F_{l-1}]$ // 将临界层之前的个体选择到 P 中;
- 4: $Q =$ 归一化 (Q);
- 5: 将 Q 中的极值点选择到 P 中;
- 6: while $|P| < N$ do
- 7: 计算未选中解到选中解的角度距离;
- 8: 确定与选中解距离最远的未选中解 p ;
- 9: $P = P \cup \{p\}, Q = Q \setminus \{p\}$;
- 10: $FrontNo = NDFrontNo(P)$;
- 11: return $P, FrontNo$.

1) 基于非支配排序的个体选择.

对混合种群 Q 中的个体进行非支配排序, 保留临界层 F_l (l 是满足 $|F_1 \cup \dots \cup F_l| \geq N$ 的最小整数) 及其之前层级的个体, 并将临界层之前层级的个体选择到 P 中. 为使种群中的个体分布广泛, 选择归一化后 Q 中的极值点至 P . 对于 $\mathbf{x}_i \in Q$, 其归一化目标向量 $\mathbf{F}'(\mathbf{x}_i) = (f'_1(\mathbf{x}_i), \dots, f'_m(\mathbf{x}_i))$ 可根据理想点 $\mathbf{z}^{\min} = (z_1^{\min}, \dots, z_m^{\min})$ 和底点 $\mathbf{z}^{\max} = (z_1^{\max}, \dots, z_m^{\max})$ 得到, 即

$$f'_j(\mathbf{x}_i) = \frac{f_j(\mathbf{x}_i) - z_j^{\min}}{z_j^{\max} - z_j^{\min}}, j = 1, \dots, m. \quad (4)$$

其中: $z_j^{\min} = \min_{i=1}^{|Q|} f_j(\mathbf{x}_i), z_j^{\max} = \max_{i=1}^{|Q|} f_j(\mathbf{x}_i)$. Q 中沿第 i 个目标轴的极值点通过最小化下列成绩标量函数 (achievement scalarizing function, ASF)^[25] 实现:

$$ASF(\mathbf{x}, \mathbf{w}_i) = \max_{j=1}^m \frac{f'_j(\mathbf{x})}{w_{i,j}}. \quad (5)$$

其中: $\mathbf{w}_i = (w_{i,1}, \dots, w_{i,m})$ 表示第 i 个目标轴的方向向量, $i = j$ 时 $w_{i,j} = 1, i \neq j$ 时 $w_{i,j} = 10^{-6}$.

2) 基于“最大化最小角度”的个体选择.

将临界层之前的个体以及极值点选择到 P 后, 若 $|P| < N$, 则按照“最大化最小角度”策略从 Q 中选择个体. “最大化最小角度”考虑种群多样性, 旨在选择与选中解距离最远的未选中解. 对于未选中解 $\mathbf{x}_q \in Q$ 与选中解 $\mathbf{x}_p \in P$, 它们之间的角度为

$$\text{Angle}(\mathbf{x}_q, \mathbf{x}_p) = \arccos\left(\frac{\mathbf{F}'(\mathbf{x}_q) \cdot \mathbf{F}'(\mathbf{x}_p)}{|\mathbf{F}'(\mathbf{x}_q)|_2 \cdot |\mathbf{F}'(\mathbf{x}_p)|_2}\right). \quad (6)$$

其中: $\mathbf{F}'(\mathbf{x}_q), \mathbf{F}'(\mathbf{x}_p)$ 表示个体的归一化目标向量, $|\mathbf{F}'(\mathbf{x}_q)|_2, |\mathbf{F}'(\mathbf{x}_p)|_2$ 表示归一化目标向量的2-范数. 将未选中解到选中解的角度中最小的一个作为该未选中解到选中解集 P 的距离, 接着选择一个距离最远的解加入 P , 并将该解从 Q 中删除. 逐一选择直至 $|P| = N$.

2.3.2 第 2 阶段环境选择

经过第 1 阶段进化之后,精英集逐渐逼近 PF,精英集中的非支配解起到类似参考向量的作用,可以通过聚类引导种群进化,此时进化转入第 2 阶段. 这一阶段的环境选择过程如算法 3 所示.

算法 3 第 2 阶段环境选择.

输入: 混合种群 Q , 精英集 EA, 种群规模 N ;
 输出: 下一代种群 P , 层级序号 FrontNo.
 1: $Q = \text{归一化}(Q)$;
 2: 计算 Q 中每个个体到精英集 EA 中每个个体的角度距离;
 3: 将 Q 中个体划分到与之距离最近的精英个体的类中;
 4: 计算 Q 中个体的目标值和;
 5: 在每类中进行适应值排序, 得到适应值层级序号 FitFrontNo;
 6: 对 Q 进行非支配排序, 得到非支配层级序号 NDFrontNo;
 7: $Q = [NDF_1, \dots, NDF_l] // l$ 是满足 $|NDF_1 \cup \dots \cup NDF_l| \geq N$ 的最小整数;
 8: if $|Q| > N$ then
 9: $Q = [\text{FitF}_1, \dots, \text{FitF}_k] // k$ 是满足 $|\text{FitF}_1 \cup \dots \cup \text{FitF}_k| \geq N$ 的最小整数;
 10: if $|Q| > N$ then
 11: $P = [\text{FitF}_1, \dots, \text{FitF}_{k-1}] //$ 将前 $k-1$ 层的个体选择到 P 中;
 12: 用“最大化最小角度”策略选择个体 // 算法 2 第 6~第 10 行;
 13: else
 14: $P = Q$;
 15: else
 16: $P = Q$;
 17: FrontNo = [NDFrontNo(P), FitFrontNo(P)];
 18: return P , FrontNo.

1) 基于聚类的进化引导.

基于参考向量进行分解的算法在处理 PF 形状不规则的优化问题时,均匀分布的参考向量无法保证最优解集中个体的均匀分布,因此需要在进化过程中调整参考向量. 与之相比,使用精英集中的非支配解引导进化可避免复杂的参考向量调整,由式(6)计算混合种群中的个体与精英个体之间的角度,按照角度相近的原则,将混合种群中的个体划分到与之距离最近的精英个体确定的类中.

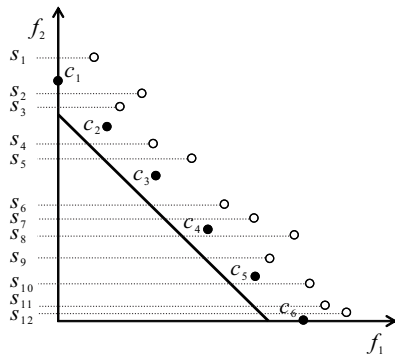


图 2 混合种群和精英集

如图 2 所示, $s_1 \sim s_{12}$ 为混合种群 Q 中的个体, $c_1 \sim c_6$ 为精英集 EA 中的个体. 图 3 所示为聚类结果, 用 $C_1 \sim C_6$ 表示精英个体 $c_1 \sim c_6$ 对应的类, $C_1 = \emptyset, C_2 = \{s_1, s_2, s_3\}, C_3 = \{s_4, s_5\}, C_4 = \{s_6, s_7\}, C_5 = \{s_8, s_9, s_{10}\}, C_6 = \{s_{11}, s_{12}\}$.

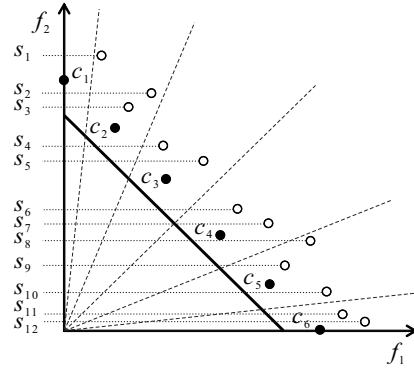


图 3 聚类

2) 基于 Pareto 支配和目标值和的个体选择.

随着优化目标个数的增多,仅靠 Pareto 支配无法有效选择个体. CSEA 根据 Pareto 支配和目标值和衡量个体的收敛性,并据此选择个体. 选择之前,在每一类中,根据个体的目标值和进行适应值排序. 对于归一化目标向量 $F'(x_i) = (f'_1(x_i), \dots, f'_m(x_i))$, 其目标值和为

$$c(x) = \sum_{i=1}^m f'_i(x). \quad (7)$$

目标值和越小的个体层级排序越低,收敛性越好. 如图 4 所示,不同颜色的虚线表示不同的适应值层级,红色最低,蓝色次之,黄色最高.

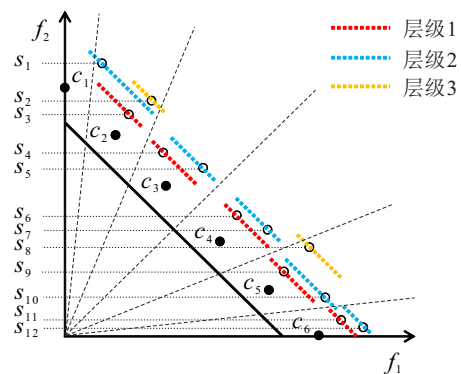


图 4 基于目标值和的适应值排序

接下来,先基于 Pareto 支配使用非支配排序选择个体,如图 5 所示. 非支配排序后, NDF_1 (NDF_i 表示 $NDF_{\text{FrontNo}} = i$ 的个体的集合) 包含 10 个非支配解, NDF_2 包含 2 个支配解 (s_2, s_8), 则临界层为 NDF_1 , Q 中保留除 s_2, s_8 外的其余 10 个个体.

最后,根据适应值排序结果以及“最大化最小角度”策略选择个体. 具体过程为: 确定满足 $|\text{FitF}_1 \cup$

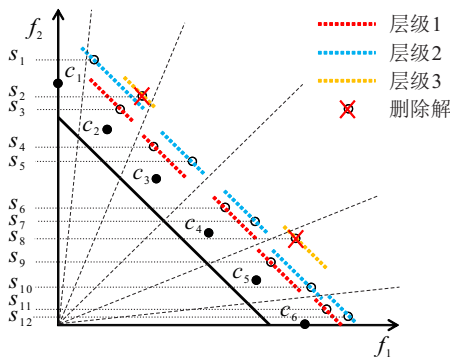


图5 基于非支配排序的个体选择

... $\cup \text{FitF}_k \geq N$ 的最小整数 k (FitF 表示 $\text{FitFrontNo} = i$ 的个体的集合), 且令 $Q = [\text{FitF}_1, \dots, \text{FitF}_k]$. 若 $|Q| = N$, 则 $P = Q$; 否则, 令 $P = [\text{FitF}_1, \dots, \text{FitF}_{k-1}]$, 并参照算法2第6~第10行, 从未选中个体中逐一选择个体加入 P 直至 $|P| = N$.

2.4 精英集更新

为保证精英集正确引导当前种群进化, CSEA 利用非支配排序和小生境策略定期更新精英集, 使其保持较好的收敛性和多样性.

更新时, 从精英集和子代种群中选择非支配解构成新的精英集, 如果非支配解的数量不超过设定的种群规模, 则结束更新; 否则, 计算新的精英集中每个非支配解的拥挤度, 逐一去除拥挤度最大的个体, 直至新的精英集满足种群规模的限制. 精英集 EA 中个体 p 的拥挤度按照下式计算:

$$D(p) = 1 - \prod_{q \in \text{EA}, q \neq p} R(p, q). \quad (8)$$

$$R(p, q) = \begin{cases} d(p, q)/r, & d(p, q) \leq r; \\ 1, & \text{otherwise.} \end{cases} \quad (9)$$

其中: q 是 EA 中除 p 外的其他个体; $d(p, q)$ 表示 p 与 q 之间的欧氏距离; r 表示小生境的半径, 根据精英集中所有个体到其 k 近邻解的平均值确定 (根据文献 [26], $k = 3$). 由式 (8) 和 (9) 可以看出, 拥挤度同时考虑了小生境中近邻解的数量和位置, 拥有较少近邻解和较远距离近邻解的个体会得到较小的拥挤度. 保留拥挤度较小的个体, 有利于保持精英集较好的多样性, 在逐一去除拥挤度最大的个体后, 可以得到具有较好收敛性和多样性的精英集, 从而引导种群进化.

2.5 计算复杂度分析

由算法1可知, CSEA 的计算复杂度主要来自3部分, 即匹配选择和交叉变异、环境选择、精英集更新. 令 m 、 N 、 D 分别表示优化目标个数、种群规模、决策变量维度. 匹配选择的复杂度为 $O(mN^2)$, 交叉

变异的复杂度为 $O(DN)$. 第1阶段环境选择中, 非支配排序在最差情况下的计算复杂度为 $O(mN^2)$, 目标归一化和“最大化最小角度”策略的复杂度分别为 $O(mN)$ 、 $O(mN^2)$; 第2阶段环境选择中, 聚类的计算复杂度为 $O(mN^2)$, 目标值和计算和适应值排序的复杂度为 $O(mN)$ 、 $O(N^2)$, 个体选择的复杂度为 $O(mN^2)$, 因此, 整个环境选择过程的计算复杂度为 $O(mN^2)$. 精英集更新的计算复杂度为 $O(mN^2)$. 因此, CSEA 总的计算复杂度为 $O(mN^2)$.

3 实验仿真与分析

3.1 对比算法、测试问题与评价指标

为验证 CSEA 的有效性, 选取7种算法 MOEA/D^[15]、NSGA-III^[25]、RVEA^[27]、MaOEA-IGD^[9]、DGEA^[28]、TSTI^[29]、TriP^[30], 借助进化多目标优化平台 PlatEMO^[31] 与本文算法进行对比实验.

从常用的高维多目标优化问题测试集中选取18个测试问题: WFG1~WFG9^[32]、MaF1~MaF9^[33], 它们包含了不同情形的 PF, 比如规则形状、不连续、退化、逆等, 可检验 CSEA 在不同类型高维多目标优化问题上的性能.

采用反转世代距离 (inverted generation distance, IGD) 和超体积 (hypervolume, HV) 两个指标评价算法性能, 这两个指标均为综合指标, 可以同时评价算法所得最优解集的收敛性和多样性, IGD 越小, 或者 HV 越大, 算法的综合性能越好.

3.2 实验参数设置

所有算法均采用模拟二进制交叉和多项式变异算子, 算子中的两个参数 $\eta_c = \eta_m = 20$, 交叉概率为1, 变异概率为 $1/D$ (D 为决策变量维度). 对比算法参数设置与原文献一致, CSEA 精英集更新频率 $f_r = 0.1$.

种群规模 N 用双层法确定, 如表1所示, 其中 H_1 、 H_2 分别是外层和内层沿目标轴的分段数.

表1 种群规模

m	分段数 (H_1, H_2)	种群规模 N
3	13, 0	105
5	10, 0	210
8	3, 2	156
10	3, 2	275
15	2, 1	135

最大迭代次数参照文献 [34] 和文献 [35] 设置, 如表2所示.

3.3 实验结果分析

各实验独立运行30次, 得到平均值和标准差. 为

表 2 最大迭代次数

<i>m</i>	3	5	8	10	15
WFG1~WFG3	52 500	105 000	156 000	275 000	202 500
WFG4~WFG9	31 500	63 000	78 000	137 500	135 000
MaF1	20 000	60 000	80 000	100 000	150 000
MaF2	20 000	40 000	60 000	70 000	100 000
MaF3	60 000	80 000	120 000	120 000	150 000
MaF4	60 000	80 000	150 000	150 000	200 000
MaF5	10 000	20 000	20 000	20 000	40 000
MaF6	20 000	40 000	40 000	40 000	60 000
MaF7	30 000	40 000	60 000	70 000	100 000
MaF8	60 000	80 000	80 000	80 000	100 000
MaF9	60 000	60 000	60 000	60 000	100 000

判断 CSEA 与对比算法的性能有无显著差异, 根据平均值进行显著性水平为 0.05 的 Wilcoxon 秩和检验. 下文表格中: “+” “-” “=” 分别表示对比算法显著优于、显著劣于、无差别于 CSEA; “rank” 表示算法在测试题集上的平均排名, 排名越小, 算法性能越好.

表 3 为各算法在 WFG1~WFG9 上的 Wilcoxon 秩和检验结果. 从 IGD 看: CSEA 显著优于 MOEA/D、MaOEA-IGD、DGEA、TriP, 显著占优的比例为 97.8%、97.8%、86.7%、44.4%; 与 RVEA、TSTI 无显著差别; 显著劣于 NSGA-III. 排名前 3 的算法分别是 NSGA-III、TSTI、CSEA. 从 HV 看, CSEA 显著优于除 NSGA-III 和 RVEA 外的其他 5 种对比算法, 显著占优的比例为 93.3%、91.1%、95.6%、60%、71.1%. 排名前 3 的算法分别是 NSGA-III、RVEA、CSEA. 综合 IGD 和 HV 来看, 在 WFG 测试题上, NSGA-III 和 RVEA 表现较好. 这是因为, WFG 的 9 个测试题中有 6 个是规则测试题 (WFG4~WFG9), 即它们的 PF 形状规则分布, 不存在不连续、退化等不规则 PF, 而在 NSGA-III 和 RVEA 中, 均匀分布的参考向量恰与这些规则测试题的 PF 分布一致, 因此 NSGA-III 和 RVEA 能取得较好

表 3 WFG 测试题 Wilcoxon 秩和检验结果

算法	IGD				HV			
	+	-	=	rank	+	-	=	rank
MOEA/D	1	44	0	6.62 (7)	0	42	3	6.49 (6)
NSGA-III	26	12	7	2.58 (1)	40	0	5	1.58 (1)
RVEA	20	20	5	3.33 (4)	22	12	11	2.91 (2)
MaOEA-IGD	0	44	1	7.93 (8)	2	41	2	7.22 (8)
DGEA	1	39	5	5.71 (6)	0	43	2	6.49 (6)
TSTI	19	19	7	3.11 (2)	17	27	1	3.22 (4)
TriP	13	20	12	3.53 (5)	9	32	4	4.18 (5)
CSEA				3.18 (3)				3.16 (3)

的结果. 但在处理 PF 不规则的优化问题时, 它们的性能会下降, 而 CSEA 仍可以保持较好的性能, 这一点将在 MaF 测试题上得到验证.

表 4 为各算法在 MaF1~MaF9 上的 Wilcoxon 秩和检验结果. 从 IGD 看, CSEA 显著优于除 TriP 外的其他 6 种对比算法, 显著占优的比例为 71.1%、51.1%、66.7%、82.2%、80%、48.9%, 虽劣于 TriP, 但差距较小. 排名前 3 的算法分别是 TriP、CSEA 和 NSGA-III. 从 HV 看, CSEA 显著优于除 NSGA-III 外的其他 6 种对比算法, 显著占优的比例为 62.2%、68.9%、82.2%、82.2%、66.7%、57.8%; 稍劣于 NSGA-III. 排名前 3 的算法分别是 NSGA-III、CSEA、TriP.

表 4 MaF 测试题 Wilcoxon 秩和检验结果

算法	IGD				HV			
	+	-	=	rank	+	-	=	rank
MOEA/D	12	32	1	5.2 (6)	13	28	4	4.29 (4)
NSGA-III	15	23	7	3.47 (3)	18	17	10	2.82 (1)
RVEA	10	30	5	4.56 (5)	9	31	5	4.4 (5)
MaOEA-IGD	7	37	1	6.44 (8)	4	37	4	6.18 (7)
DGEA	5	36	4	6.09 (7)	4	37	4	6.27 (8)
TSTI	19	22	4	3.8 (4)	6	30	9	4.53 (6)
TriP	22	19	4	3.16 (1)	11	26	8	4.2 (3)
CSEA				3.29 (2)				3 (2)

与 WFG 测试题相比, NSGA-III 和 RVEA 在 MaF 测试题上性能均出现下降, 这是因为, MaF 各测试题有较复杂的 PF, NSGA-III 和 RVEA 中均匀分布的参考向量与这些复杂的 PF 形状分布不完全一致. CSEA 不使用参考向量, 而是利用收敛性和多样性较好的、储存非支配解的精英集引导当前种群进化, 可根据非支配解逼近的 PF 形状确定进化方向, 从而避免均匀分布的参考向量与不规则 PF 形状分布不一致对最终结果的影响, 使 CSEA 能够处理不同类型的高维多目标优化问题.

为进一步分析各算法在不同优化目标个数上的性能, 将各算法在同一目标个数测试题上的排名取平均值, 如图 6 所示.

由图 6(a)、(b) 知, 在 WFG 测试题上, CSEA 在 10 目标和 15 目标测试题上的性能好于 3 目标、5 目标、8 目标测试题. 由图 6(c)、(d) 知, 在 MaF 测试题上, CSEA 在各目标测试题上均有较好表现.

通过算法整体性能和算法在不同目标个数上的性能两方面对各算法进行对比分析, 可得出以下结论:

- 1) 无论优化问题的 PF 是规则的还是不规则的, CSEA 均能取得较好的效果;

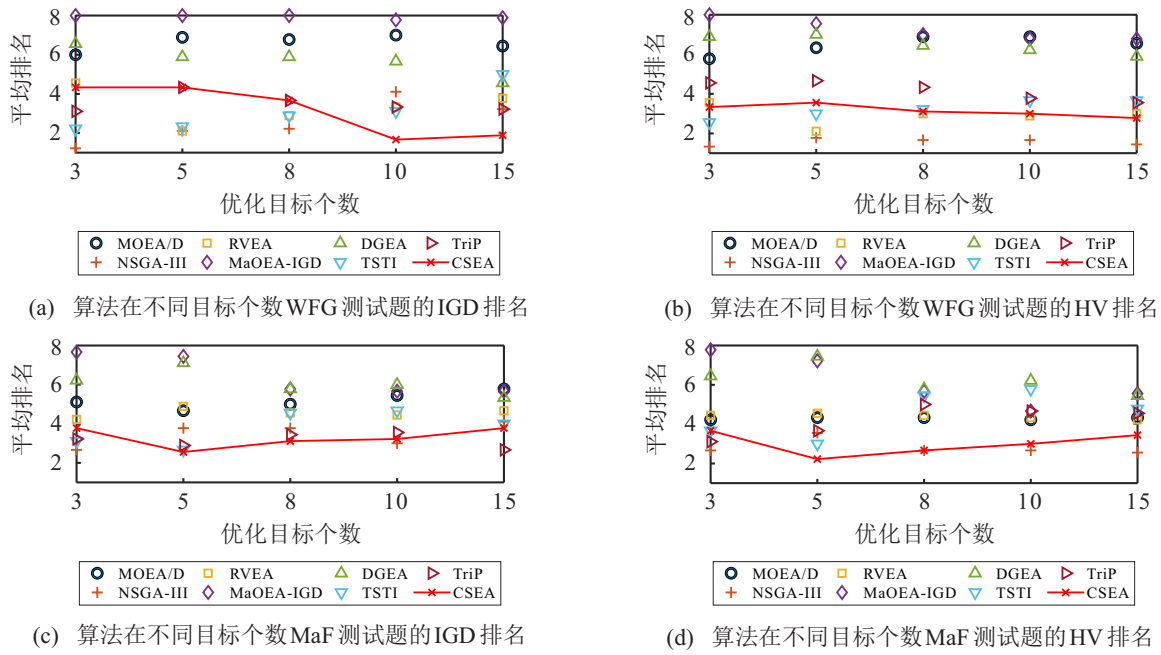


图6 算法在不同目标个数测试题的平均排名

2) 相比于具有规则前沿的优化问题(如WFG4~WFG9),CSEA更擅长处理具有不规则前沿的优化问题;

3) 在MaF测试题和HV评价指标下的WFG测试题上,CSEA在不同目标个数上的排名曲线无较大波动,说明CSEA性能稳定,可以求解不同目标个数的优化问题.

3.4 策略有效性分析

为分析CSEA中聚类引导和目标值和这两种策略的有效性,将CSEA与两种变体算法进行对比,其中一种变体算法记为CSEA-nC,用均匀分布的参考向量引导进化代替CSEA中精英集中的非支配解引导进化;另一种变体算法记为CSEA-nS,用随机选择代替CSEA中基于目标值和的适应值排序选择.变体算法其他方面与CSEA相同,实验设置与前述相同,对算法求得解集的IGD和HV进行Wilcoxon秩和检验,结果如表5所示.

表5 CSEA及其变体算法Wilcoxon秩和检验结果

测试题	vs.CSEA	IGD			HV		
		+	-	=	+	-	=
WFG	CSEA-nC	9	31	5	21	9	15
	CSEA-nS	9	29	7	2	41	2
MaF	CSEA-nC	11	24	10	15	21	9
	CSEA-nS	22	16	7	14	20	11

从IGD角度,在WFG上,CSEA-nC和CSEA-nS显著劣于CSEA,说明两个策略均有效.在MaF上,

CSEA-nC显著劣于CSEA,而CSEA-nS显著优于CSEA,说明对具有复杂PF形状的MaF测试题而言,聚类引导策略有效,但基于目标值和的适应值排序选择不如随机选择效果好,这可能是对具有复杂PF形状的优化问题而言,目标值和的等高线与复杂PF的形状(尤其是非线性PF)不相符,不能准确反映个体的收敛性.

从HV角度,在MaF上,CSEA-nC和CSEA-nS显著劣于CSEA,说明两个策略均有效.在WFG上,CSEA-nS显著劣于CSEA,而CSEA-nC显著优于CSEA,说明对WFG测试题而言,基于目标值和的适应值排序选择策略有效,但聚类引导效果欠佳,其原因与NSGA-III和RVEA在WFG测试题上表现较好的原因类似.CSEA-nC采用均匀分布的参考向量,其分布与WFG4~WFG9等6个规则测试题的PF形状一致,因此,采用均匀分布的参考向量引导进化比采用精英集中的非支配解引导进化效果好.

4 结论

本文提出了一种基于聚类引导和目标值和高维多目标进化算法CSEA.为避免处理不规则前沿优化问题时调整参考向量,CSEA利用一个储存非支配解的精英集引导种群进化.为使精英集正确引导种群进化,整个进化过程被分为两个阶段.第1阶段不使用精英集进行环境选择,而使用非支配排序和“最大化最小角度”策略.经过第1阶段进化后,非支配解逐渐逼近Pareto前沿,可以反映Pareto前沿形状分布,因此,在第2阶段使用精英集通过聚类引导种群进化,

并利用非支配排序、基于目标值和的适应值排序以及“最大化最小角度”策略选择个体。此外, CSEA利用非支配排序和小生境策略定期更新精英集, 使其保持较好的收敛性与多样性, 避免了对种群进化产生误导作用。为验证CSEA的有效性, 选取7种对比算法, 在WFG1~WFG9、MaF1~MaF9共18个高维多目标优化测试题上进行了实验, 结果表明, CSEA可以有效解决不同类型的高维多目标优化问题。

策略有效性分析实验表明, 在某些具有复杂PF形状的优化问题上, 目标值和不能准确反映个体的收敛性。基于此, 后续研究将考虑根据PF形状, 选择相应的收敛性度量方式, 其中的关键是如何高效准确地判断PF形状。另外, 大多数现有算法难以同时有效解决规则优化问题和不规则优化问题, 而能同时解决这两类问题的算法效率又有待提升。因此, 在不掌握优化问题先验知识的前提下, 即不知道待优化问题的PF规则与否, 如何选择与设计优化算法也是值得思考的问题。

参考文献(References)

- [1] Li B D, Li J L, Tang K, et al. Many-objective evolutionary algorithms[J]. *ACM Computing Surveys*, 2015, 48(1): 1-35.
- [2] Zhou A M, Qu B Y, Li H, et al. Multiobjective evolutionary algorithms: A survey of the state of the art[J]. *Swarm and Evolutionary Computation*, 2011, 1(1): 32-49.
- [3] 刘建昌, 李飞, 王洪海, 等. 进化高维多目标优化算法研究综述[J]. *控制与决策*, 2018, 33(5): 879-887.
(Liu J C, Li F, Wang H H, et al. Survey on evolutionary many-objective optimization algorithms[J]. *Control and Decision*, 2018, 33(5): 879-887.)
- [4] 肖人彬, 李贵, 陈峙臻. 进化超多目标优化研究进展及展望[J]. *控制与决策*, 2023, 38(7): 1761-1788.
(Xiao R B, Li G, Chen Z Z. Research progress and prospect of evolutionary many-objective optimization[J]. *Control and Decision*, 2023, 38(7): 1761-1788.)
- [5] Das S S, Islam M M, Arafat N A. Evolutionary algorithm using adaptive fuzzy dominance and reference point for many-objective optimization[J]. *Swarm and Evolutionary Computation*, 2019, 44: 1092-1107.
- [6] 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.
- [7] Tian Y, Cheng R, Zhang X Y, et al. A strengthened dominance relation considering convergence and diversity for evolutionary many-objective optimization[J]. *IEEE Transactions on Evolutionary Computation*, 2019, 23(2): 331-345.
- [8] 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.
- [9] Sun Y N, Yen G G, Yi Z. IGD indicator-based evolutionary algorithm for many-objective optimization problems[J]. *IEEE Transactions on Evolutionary Computation*, 2019, 23(2): 173-187.
- [10] Bader J, Zitzler E. HypE: An algorithm for fast hypervolume-based many-objective optimization[J]. *Evolutionary Computation*, 2011, 19(1): 45-76.
- [11] Jiang S W, Zhang J, Ong Y S, et al. A simple and fast hypervolume indicator-based multiobjective evolutionary algorithm[J]. *IEEE Transactions on Cybernetics*, 2015, 45(10): 2202-2213.
- [12] 李飞, 吴紫恒, 刘澜蓉, 等. 基于R2指标和目标空间分解的高维多目标粒子群优化算法[J]. *控制与决策*, 2021, 36(9): 2085-2094.
(Li F, Wu Z H, Liu K R, et al. R2 indicator and objective space partition based many-objective particle swarm optimizer[J]. *Control and Decision*, 2021, 36(9): 2085-2094.)
- [13] 陈国玉, 李军华, 黎明, 等. 基于R2指标和参考向量的高维多目标进化算法[J]. *自动化学报*, 2021, 47(11): 2675-2690.
(Chen G Y, Li J H, Li M, et al. An R2 indicator and reference vector based many-objective optimization evolutionary algorithm[J]. *Acta Automatica Sinica*, 2021, 47(11): 2675-2690.)
- [14] Trivedi A, Srinivasan D, Sanyal K, et al. A survey of multiobjective evolutionary algorithms based on decomposition[J]. *IEEE Transactions on Evolutionary Computation*, 2017, 21(3): 440-462.
- [15] 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.
- [16] Liu H L, Gu F Q, Zhang Q F. Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems[J]. *IEEE Transactions on Evolutionary Computation*, 2014, 18(3): 450-455.
- [17] Qi Y T, Ma X L, Liu F, et al. MOEA/D with adaptive weight adjustment[J]. *Evolutionary Computation*, 2014, 22(2): 231-264.
- [18] Li M Q, Yao X. What weights work for you?

- Adapting weights for any Pareto front shape in decomposition-based evolutionary multiobjective optimization[J]. *Evolutionary Computation*, 2020, 28(2): 227-253.
- [19] Wang Z K, Zhang Q F, Li H, et al. On the use of two reference points in decomposition based multiobjective evolutionary algorithms[J]. *Swarm and Evolutionary Computation*, 2017, 34: 89-102.
- [20] Ma X L, Yu Y N, Li X D, et al. A survey of weight vector adjustment methods for decomposition-based multiobjective evolutionary algorithms[J]. *IEEE Transactions on Evolutionary Computation*, 2020, 24(4): 634-349.
- [21] Hong R, Yao F, Liao T J, et al. Growing neural gas assisted evolutionary many-objective optimization for handling irregular Pareto fronts[J]. *Swarm and Evolutionary Computation*, 2023, 78: 101273.
- [22] Wang M J, Li X P, Dai Y, et al. An incremental learning evolutionary algorithm for many-objective optimization with irregular Pareto fronts[J]. *Information Sciences*, 2023, 642: 119115.
- [23] Zhang X Y, Tian Y, Cheng R, et al. An efficient approach to nondominated sorting for evolutionary multiobjective optimization[J]. *IEEE Transactions on Evolutionary Computation*, 2015, 19(2): 201-213.
- [24] Xiang Y, Zhou Y R, Li M Q, et al. A vector angle based evolutionary algorithm for unconstrained many-objective problems[J]. *IEEE Transactions on Evolutionary Computation*, 2017, 21(1): 131-152.
- [25] 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.
- [26] Li M Q, Yang S X, Liu X H. Pareto or non-Pareto: Bi-criterion evolution in multiobjective optimization[J]. *IEEE Transactions on Evolutionary Computation*, 2016, 20(5): 645-665.
- [27] Cheng R, Jin Y C, Olhofer M, et al. A reference vector guided evolutionary algorithm for many-objective optimization[J]. *IEEE Transactions on Evolutionary Computation*, 2016, 20(5): 773-791.
- [28] He C, Cheng R, Yazdani D. Adaptive offspring generation for evolutionary large-scale multiobjective optimization[J]. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2022, 52(2): 786-798.
- [29] Dong J, Gong W Y, Ming F, et al. A two-stage evolutionary algorithm based on three indicators for constrained multi-objective optimization[J]. *Expert Systems with Applications*, 2022, 195: 116499.
- [30] Ming F, Gong W Y, Wang L, et al. A tri-population based co-evolutionary framework for constrained multi-objective optimization problems[J]. *Swarm and Evolutionary Computation*, 2022, 70: 101055.
- [31] 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.
- [32] 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.
- [33] 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.
- [34] Xiang Y, Zhou Y R, Yang X W, et al. A many-objective evolutionary algorithm with Pareto-adaptive reference points[J]. *IEEE Transactions on Evolutionary Computation*, 2020, 24(1): 99-113.
- [35] Xu H, Zeng W H, Zeng X X, et al. A polar-metric-based evolutionary algorithm[J]. *IEEE Transactions on Cybernetics*, 2021, 51(7): 3429-3440.

作者简介

王旭健(1994—), 男, 博士生, 从事多目标优化、阵列天线优化设计等研究, E-mail: wxj_903@163.com;

张峰干(1985—), 男, 副教授, 博士, 从事阵列信号处理、动中通系统优化设计等研究, E-mail: zfg417@163.com;

姚敏立(1966—), 男, 教授, 博士生导师, 从事阵列信号处理、动中通系统优化设计等研究, E-mail: yaominli66@163.com.