

控制与决策

Control and Decision

基于多区域中心点预测的动态多目标优化算法

马学敏, 杨景明, 孙浩, 呼子宇, 韦钦楠

引用本文:

马学敏, 杨景明, 孙浩, 呼子宇, 韦钦楠. 基于多区域中心点预测的动态多目标优化算法[J]. 控制与决策, 2022, 37(10): 2477–2486.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2021.0268>

您可能感兴趣的其他文章

Articles you may be interested in

基于黑洞多目标进化算法的永磁直线同步电机优化设计

Optimization of a tubular coreless linear PM synchronous machine based on multi-objective blackhole evolutionary algorithm
控制与决策. 2022, 37(6): 1567–1572 <https://doi.org/10.13195/j.kzyjc.2020.0869>

基于分类的多策略预测方法求解动态多目标优化问题

Classification-based multi-strategy prediction method for dynamic multi-objective optimization problems
控制与决策. 2021, 36(7): 1569–1580 <https://doi.org/10.13195/j.kzyjc.2019.1320>

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>

基于向量角分解的高维多目标进化算法

Many-objective evolutionary algorithm based on vector angle decomposition
控制与决策. 2021, 36(3): 761–768 <https://doi.org/10.13195/j.kzyjc.2019.0925>

基于弱关联的自适应高维多目标进化算法

A weak association-based adaptive evolutionary algorithm for many-objective optimization
控制与决策. 2021, 36(8): 1804–1814 <https://doi.org/10.13195/j.kzyjc.2019.1723>

基于多区域中心点预测的动态多目标优化算法

马学敏, 杨景明, 孙浩[†], 呼子宇, 韦钦楠

(1. 燕山大学 电气工程学院, 河北 秦皇岛 066004;
2. 燕山大学 智能控制系统与智能装备教育部工程研究中心, 河北 秦皇岛 066004)

摘要: 现实生活中存在很多动态多目标优化问题 (DMOPs), 这类问题要求算法在环境变化后快速收敛到新的 Pareto 前沿, 并保持解集的多样性. 随着 Pareto 前沿复杂程度的增加, 这一问题更加突出. 鉴于此, 提出一种基于多区域中心点预测的动态多目标优化算法 (MCPDMO). 首先, 根据环境变化的严重程度将种群划分为多个子区域, 使得个体的分配更加适应动态变化的环境; 然后, 分别计算每个子区域的中心点, 对不同子区域在不同时刻的中心点建立时间序列, 并利用差分模型预测新环境的最优解集, 以提高算法对不同环境变化的响应能力; 最后, 为验证算法的有效性, 与 3 种动态多目标优化算法在 10 个标准测试函数上进行仿真实验. 实验结果表明, 所提出算法在具有复杂 Pareto 前沿的动态问题上表现出更优的收敛性和分布性.

关键词: 动态多目标优化; 多目标优化; 进化算法; 预测; 环境变化程度; 子区域

中图分类号: TP273 文献标志码: A

DOI: 10.13195/j.kzyjc.2021.0268

引用格式: 马学敏, 杨景明, 孙浩, 等. 基于多区域中心点预测的动态多目标优化算法 [J]. 控制与决策, 2022, 37(10): 2477-2486.

Dynamic multi-objective optimization algorithm based on multi-regional center point prediction

MA Xue-min, YANG Jing-ming, SUN Hao[†], HU Zi-yu, WEI Qin-nan

(1. School of Electrical Engineering, Yanshan University, Qinhuangdao 066004, China; 2. Engineering Research Center of the Ministry of Education for Intelligent Control System and Intelligent Equipment, Yanshan University, Qinhuangdao 066004, China)

Abstract: In real life, there are many dynamic multi-objective optimization problems (DMOPs), which require algorithms to quickly converge to the new Pareto front and maintain the diversity of the solution set. As the complexity of the Pareto front increases, this problem is more prominent. For this problem, a dynamic multi-objective optimization algorithm based on multi-regional center point prediction (MCPDMO) is proposed. Firstly, the population is divided into multiple sub-regions according to the degree of environmental change, which makes the allocation of individuals more adaptable to the dynamic environment. Then, the center point of each sub-region is calculated respectively. The time series are established with the center point of different sub-regions at different times, and the difference model is used to predict the optimal solution set of the new environment, which improve the algorithm's ability to respond to changes in different environments. To verify the effectiveness of the algorithm, the simulation experiments are conducted on 10 test functions with 3 dynamic multi-objective optimization algorithms. The results show that the algorithm exhibits better convergence and distribution in dynamic problems with complex Pareto front.

Keywords: dynamic multi-objective optimization; multi-objective optimization; evolutionary algorithm; prediction; degree of environmental change; sub-region

0 引言

动态多目标优化问题 (dynamic multi-objective optimization problems, DMOPs) 广泛存在于现实生活

中, 如生产调度、资源分配和组合优化领域^[1-2]. 因此, 解决动态多目标优化问题具有重要的实际意义. DMOPs 是指其目标函数不仅与决策变量有关, 而且

收稿日期: 2021-02-09; 录用日期: 2021-07-05.

基金项目: 国家自然科学基金项目 (62003296); 河北省自然科学基金项目 (F2020203031); 河北省教育厅科技项目 (QN2020225); 国家重点研发计划项目 (2018YFB1702300).

责任编辑: 巩敦卫.

[†]通讯作者. E-mail: sunhao@ysu.edu.cn.

还会随着时间(环境)动态变化,其最优解也会随着时间(环境)动态变化^[3]. 与求解静态多目标优化问题相比,DMOPs需要在环境变化后找到近似Pareto最优解集(PS)和Pareto前沿(PF),这便要求算法能够及时跟踪随时间变化的环境^[4].

近年来,在动态多目标优化算法领域,学者提出了很多动态多目标优化算法(dynamic multi-objective optimization algorithms, DMOAs)^[5]. 现有的DMOAs可分为3类:多样性策略、记忆机制以及预测策略.

1) 多样性策略. 该方法通过多样性引入多样性维持机制增加种群的多样性. Zou等^[6-7]提出了一种动态演化环境模型解决环境变化时的DMOP,该模型利用环境信息指导不同子种群的进化,以增强种群的多样性. 但当环境严重程度变化较大时,历史环境信息不能很好地引导新环境. Liu等^[8]提出了一种改进的协同进化多种群粒子群优化算法(CMPsODMO),当环境发生变化时,通过重新初始化一部分个体扩大搜索空间,但该方法计算复杂度大. Jiang等^[9]提出了一种稳态和分代进化算法(SGEA),当环境发生变化时,重新加载一部分分布性良好的保留解和接近Pareto真实前沿的解,但使用随机解导致种群整体沿前沿扩散,很难保持获得解的均匀性. Ruan等^[10]提出了一种混合多样性维持策略,在预测的下一时刻PS的可能范围随机产生一些个体增加种群多样性,但该方法在前期收敛速度较慢.

2) 记忆机制. 基于记忆策略方法的基本思想是利用先前搜索的最优解加速收敛进程,对于周期性问题效果较好. 为了更好地预测环境变化,一些学者对基于记忆策略方法进行了深入研究^[11]. Peng等^[12]提出了一种新的记忆和预测策略(PMS)检测环境变化并预测种群个体的新位置,PMS通过重新使用之前环境保存的精英解指导种群进化,即使算法性能明显提升,但当种群尚未完全收敛时会影响种群进化方向的判断. Azzouz等^[13]提出了一种基于记忆策略的种群管理策略,充分利用之前环境的最优解加速种群快速收敛,基于记忆策略的方法可以很好地应对周期性变化问题,但环境变化程度较大时,之前环境的信息可能会对新种群的预测产生误导.

3) 预测策略. 基于预测的方法通过每次环境变化后的预测机制指导种群进化,初始化新环境的种群^[14-16],Zhou等^[17]提出了一种种群预测策略(PPS). 最优解集分为两部分:种群中心点和流形. 通过记录连续时间序列历史种群中心点,利用AR模型对下一时刻种群的中心点和流形进行预测. 因此,

该算法的时间复杂度较小,但缺乏对PS变化剧烈问题的适应性,在前期经验不足时效果较差. Zou等^[18]提出一种基于中心点及拐点预测的动态优化方法(CKPS),该方法在环境发生变化后仅预测种群的中心点和拐点,而不是整个种群,加快了种群的收敛速度,减少了计算复杂度. 同时提出一种自适应多样性维持策略保持种群多样性,但由于前期经验不足,在三目标优化问题上仍有较大提升空间. Ma等^[19]提出一种基于特殊点的迁移学习(FIP)方法寻找环境变化后的前沿,该方法可获得收敛性较好的解集,但时间计算复杂度较大. Cao等^[20]提出了一种基于支持向量回归(SVR)的方法生成新环境的初始种群,每个子问题均可以在不同的环境中获得其相应的解,以形成时间序列. 与其他方法相比,通过SVR模型预测新环境下的子问题对解决DMOPs取得了较好的效果,但仍然存在对快速变化的环境跟踪能力较差、解集分布不均匀等问题. 而且对于变化复杂的前沿,仅仅预测PS的中心点无法充分捕捉PS的变化. 并且在预测的PS中心点附近产生个体作为新环境的初始种群效率不高. 因此,有必要将种群划分为多个子区域.

本文基于上述分析,综合考虑变化复杂的PF或PS追踪、种群多样性等问题,提出一种基于多区域中心点预测的动态多目标优化算法(dynamic multiobjective optimization evolutionary algorithm based on multiregional center points, MCPDMO). 该算法根据环境变化严重程度将种群划分至多个子区域,分别计算每个子区域的中心点. 将同一个子区域的个体分别建立时间序列,对新环境下的PS进行预测,能够保证解集均匀分布性的前提下增加预测精度. 所提出算法的特点是:1)根据环境变化严重程度自适应计算分子区域的个数;2)利用参考向量的结构化分布增加种群多样性;3)针对无关联个体的区域,提出有效的处理方式. 采用动态标准测试问题对算法性能进行仿真测试,实验结果表明,所提出算法在大多数测试函数上得到的结果优于其他算法,从而表明了MCPDMO算法的有效性.

1 背景知识

1.1 动态多目标优化问题

一般地,动态多目标优化问题定义如下:

$$\begin{aligned} \min : & \mathbf{F}(\mathbf{x}, t) = \{f_1(\mathbf{x}, t), f_2(\mathbf{x}, t), \dots, f_m(\mathbf{x}, t)\}. \\ \text{s.t. } & g_i(\mathbf{x}, t) \leq 0, \quad i = 1, 2, \dots, p; \\ & h_j(\mathbf{x}, t) = 0, \quad j = 1, 2, \dots, q. \end{aligned} \quad (1)$$

其中: t 为离散时间序列, $t = (1/n_t)[(\tau)/\tau_t]$, n_t 、 τ 、 τ_t 分别为变化的程度、迭代次数和变化的频率; $\mathbf{x} =$

$(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n)$ 为 n 维决策向量且属于决策空间 Ω ; m 为目标空间的维数; $\mathbf{F} = (f_1, f_2, \dots, f_m)$ 为被优化的 m 维目标向量; p 和 q 分别为等式约束和不等式约束的数量.

定义1 (Pareto 支配) 设 p 和 q 为种群中任意2个个体, p 支配 q , 当且仅当 $f_i(p) \leq f_i(q), \forall i=1, 2, \dots, m$, 存在 $f_j(p) < f_j(q), j=1, 2, \dots, m$, 表示为 $f(p) \prec f(q)$.

定义2 (Pareto 最优解集, PS) 设 \mathbf{x} 为决策变量, Ω 为决策空间, \mathbf{F} 为目标函数, 则 PS 定义为 $\{\mathbf{PS} = \mathbf{x} \in \Omega | \neg \exists \mathbf{x}^* \in \Omega, \mathbf{F}(\mathbf{x}^*) \prec \mathbf{F}(\mathbf{x})\}$.

定义3 (Pareto 最优前沿, PF) 设 \mathbf{x} 为决策变量, $\mathbf{F}(\mathbf{x})$ 为目标函数, 则 PF 定义为

$$\text{PF} = \{y = \mathbf{F}(\mathbf{x}) | \mathbf{x} \in \text{PS}\}.$$

基于决策空间中 PS 和目标空间中 PF 的变化, DMOPs 可以分为以下4种类型:

- 1) PS 随时间变化, PF 不随时间变化;
- 2) PS 和 PF 均随时间变化;
- 3) PS 不随时间变化, PF 随时间变化;
- 4) 问题环境发生变化, PS 和 PF 均不随时间变化.

在现实生活中, 当环境变化时, 这4种类型的变化可能同时发生, 主要关注前3种变化类型.

1.2 基于分解的多目标优化算法

基于分解的多目标进化算法 (multi-objective evolutionary algorithm based on decomposition, MOEA/D)^[21] 将多目标优化问题转化为一系列单目标优化子问题, 利用一定数量相邻问题的信息, 采用进化算法对这些子问题同时进行优化^[22-23]. 本文静态算法优化器采用 MOEA/D 算法, 其中分解方法采用切比雪夫分解方法, 该方法生成1组均匀的权重向量, 将多目标优化问题划分为1组单目标子问题, 根据聚合函数值更新所在邻域内的劣解, 找到1组最优 Pareto 解集. 由于分解操作的存在, 该方法在保持解的分布性方面有很大优势, 通过分析相邻问题的信息优化能够避免陷入局部最优.

1.3 基于中心点的预测方法

基于中心点预测的动态多目标优化方法近年来受到学者的广泛关注. 通过记录上一时刻中心点的位置预测下一时刻的中心点, 在预测的 PS 中心点周围生成新的个体初始化新环境的种群. t 时刻 PS 的中心点定义如下:

$$C_i^t = \frac{1}{|\text{PS}^t|} \sum_{\mathbf{x} \in \text{PS}^t} \mathbf{x}_i^t. \quad (2)$$

其中: PS^t 为种群在 t 时刻的解集; $|\text{PS}^t|$ 为 PS^t 的基; \mathbf{x}_i^t 为 PS^t 中的第 i 个解. 基于中心点预测的方法通过不

同时刻的中心点计算进化方向作为种群所有个体的进化方向, 但是不同个体的进化方向是不完全相同的. 所以, 只预测1个中心点不能充分描述整个种群 PS 的变化, 为不同的个体划分不同的区域对其分别进行预测是很有必要的.

2 多区域中心点预测算法(MCPDMO)

本文根据环境变化严重程度将个体划分至不同子区域, 并通过预测不同子区域的中心点对种群进行预测, 提出一种多区域中心点预测的动态多目标优化算法(MCPDMO). 该算法主要包括划分子区域、个体关联、确定进化步长和多区域中心点预测策略等步骤, 具体如下.

2.1 划分子区域

为了将个体划分到不同的子区域, 首先需要确定子区域的个数. 本文提出根据环境变化严重程度计算子区域的个数, 将个体划分至不同区域. 采用文献[24]的方法计算环境变化严重程度, 有

$$\delta(t) = \frac{1}{M} \frac{1}{N} \sum_{j=1}^M \sum_{i=1}^N |\Delta f_j(\text{Ar}_i) - \mu_j(g)|. \quad (3)$$

其中: $\delta(t)$ 为 t 时刻环境变化严重程度; M 为目标个数; N 为种群中个体数目; $\Delta f_j(\text{Ar}_i) = [(f_j(\text{Ar}_i, g) - f_j(\text{Ar}_i, g-1)) / (\mu_j(g) - l_i(g))]; \mu_j(g) = \frac{1}{N} \sum_{i=1}^N [(f_j(\text{Ar}_i, g) - f_j(\text{Ar}_i, g-1)) / (\mu_j(g) - l_j(g))]; \Delta f_j(\text{Ar}_i)$ 为第 g 次迭代中第 i 个体第 j 维的目标函数值; $\mu_j(g)$ 和 $l_i(g)$ 为第 g 次迭代中第 j 维目标的最大值和最小值.

根据计算得到环境变化的严重程度可以确定划分子区域的个数 k , 环境变化越剧烈, 划分子区域的个数越多, 反之亦然. 子区域个数 k 的计算如下式:

$$k = \lceil k_1 + \delta(t) \times (k_2 - k_1) \rceil. \quad (4)$$

其中: k_1 和 k_2 分别为 k 的下界和上界, $\delta(t)$ 为 t 时刻的环境变化严重程度. 若 k_1 太小, 则划分的子区域数目不足以充分描述 Pareto 前沿, 因此令 $k_1 = M + 1$. 本文中 $k_2 = 2M$. 对 PS/PF 进行预测的基础是合理挑选历史信息. 本文根据环境变化严重程度确定历史解和预测解的个数, 预测解的个数为 $\delta \times N$, 剩余解为环境变化前的历史解. 预测解和历史解构成环境变化时的初始解种群.

2.2 个体关联

由于子区域的个数与参考向量的个数相同, 根据环境变化严重程度确定子区域个数后, 参考向量的个数也由此确定. 通过 $N(t)$ 个参考向量将 MOP 划分为 $N(t)$ 个标量优化子问题. 令 $\lambda(t) = \lambda^1(t), \lambda^2(t), \dots,$

$\lambda^{N(t)}(t)$ 为 $R^{m(t)}$ 一组均匀分布的权重向量, 权重向量的产生通过文献[25]得到. 子区域 Δ_i^t 的定义如下:

$$\Delta_i^t = \{F(x, t) \in R^{m(t)} \mid \langle F(x, t), \lambda^i(t) \rangle \leq \langle F(x, t), \lambda^j(t) \rangle\}. \quad (5)$$

其中: $j \in 1, 2, \dots, N(t)$, $\langle F(x, t), \lambda(t) \rangle$ 为 $F(x, t)$ 与参考线的垂直距离. 子区域确立后, 种群中的每个解均会与特定的参考向量相关联, 对于每个解 x , 相关联的子区域的索引号计算如下:

$$h = \arg \min_{i \in \{1, 2, \dots, N(t)\}} \langle \bar{F}(x, t), \lambda^i(t) \rangle, \quad (6)$$

其中 $\bar{F}(x, t)$ 为解 x 归一化后的目标向量. 该解归一化后的第 i 维目标函数值计算为

$$\bar{f}_i(x, t) = \frac{f_i(x, t) - z_i^*(t)}{z_i^{\text{nad}}(t) - z_i^*(t)}. \quad (7)$$

其中: $i \in \{1, 2, \dots, m(t)\}$; $f_i(x, t)$ 为第 i 维目标函数值; $z_i^*(t) = \min\{f_i(x, t) \mid x \in \text{PS}\}$; $z_i^{\text{nad}}(t) = \max\{f_i(x, t) \mid x \in \text{PS}\}$. 确定个体与权重向量是否关联时, 只需通过式(7)计算每个解归一化后的目标向量, 并计算该目标向量到每个权重向量的欧氏距离, 距离最小的权重向量即为个体所关联向量, 如式(6)所示. 权重向量的索引即为个体所关联子区域的索引. 由此, 每个子区域的中心点可以确定.

2.3 确定进化步长

假设每个子区域的个体与该区域的中心点有相同的进化趋势, 通过之前时刻中心点的位置确定进化步长. 同时考虑计算成本和预测精度, 进化步长 Δc_i^t 由之前两个时刻每个子区域的中心点确定. 令 $C^t = \{c_1^t, c_2^t, \dots, c_k^t\}$, $C^{t-1} = \{c_1^{t-1}, c_2^{t-1}, \dots, c_k^{t-1}\}$ 分别为 t 时刻和 $t-1$ 时刻中心点解集, 因此进化步长 Δc_i^t 为

$$\Delta c_i^t = c_i^t - c_i^{t-1}, \quad i = 1, 2, \dots, k, \quad (8)$$

其中 k 为子区域的个数, 每个子区域中心点的进化方向由该区域的个体确定.

2.4 多区域中心点预测

通过 t 时刻中心点 C^t 的步长预测 $t+1$ 时刻其他个体的位置, 有

$$x_i^{t+1} = x_i^t + \Delta c_i^t. \quad (9)$$

其中: $i = 1, 2, \dots, k$, x_i^t 为第 t 次环境变化获得的解, x_i^{t+1} 为第 $t+1$ 次环境变化的预测解. 由于种群中的 N 个个体被划分到 k 个子区域, 每个子区域的个体根据其所在子区域中心点的进化方向引导进化, 产生 $t+1$ 时刻的新个体, 如式(9)所示. 因此, 所有子区域的预测解即为环境变化后的 N 个新的候选解. 但是, 可能会出现某个权重向量无个体关联的现象, 即该区域没

有中心点. 这种情况下, 该区域的进化方向由相邻区域的个体计算得到.

由于要划分的子区域的数量是根据环境变化的严重程度确定的, 可能会出现没有个体与特定的权重向量相关联的现象, 如图1所示. 通常, 将发生两种情况: 1) 边界子区域没有关联个体; 2) 中间子区域中没有个体. 相邻个体的进化差异小于其他个体, 因此可以通过相邻子区域计算空白子区域的进化方向. 若出现空白区域相邻区域仍为空的情况, 则继续寻找该空白区域的相邻区域, 直至找到不为空的子区域. 若第 i 空白子区域为上边界区域, 则寻找第 $i+2$ 个相邻区域; 若第 i 空白子区域为下边界区域, 则寻找第 $i-2$ 个相邻区域; 若空白子区域为中间区域, 则寻找空白子区域的两侧子区域, 直至找到

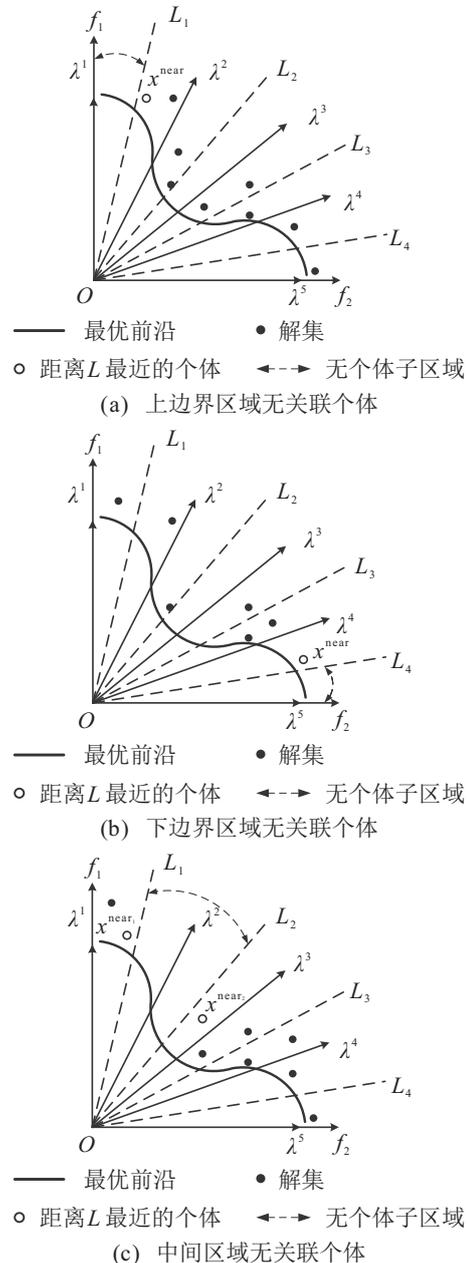


图1 子区域无关联个体情况

不为空的子区域. 因此, 对于没有解的子区域, 首先确定对应的边界 L , 然后寻找最接近 L 的个体 x^{near} . 在此测量欧几里德距离. 若空白区域位于边界区域, 则只需要1个 x^{near} 能够确定进化方向. 但是, 若空白区域位于中间区域, 则存在2个相邻边界. 因此, 需要确定2个最近的个体 x^{near_1} 和 x^{near_2} . 进化步长 Δc_i^t 定义如下:

1) 若空白区域位于边界区域, 则

$$\Delta c_i^t = F_p \times (x_t^{\text{near}} - x_{t-1}^{\text{near}}); \quad (10)$$

2) 若空白区域位于中间区域, 则

$$\Delta c_i^t = F_p \times [(x_t^{\text{near}_1} - x_{t-1}^{\text{near}_1}) + (x_t^{\text{near}_2} - x_{t-1}^{\text{near}_2})], \quad (11)$$

其中 F_p 为步长调节参数, 本文设为 $F_p = 0.4$, 参数确定过程在实验部分给出. 此外, 在某些子区域中没有个体的情况可能发生在当前代或上一代. 产生新个体时, 对于不同的个体有不同的方法, 通常分为以下3种情况:

①若本代有个体, 上一代无个体, 则由式(9)得到.

②若本代无个体, 上一代有个体, 则利用 $t-1$ 时刻个体作为初始个体产生解, $t+1$ 时刻其他个体位置为

$$x_i^{t+1} = x_i^{t-1} + \Delta c_i^t. \quad (12)$$

③若本代和上一代均无个体, 则无需识别没有个体的分区, 此时会发生两种情况: 边界分区或中间分区中没有个体. 然后, 时间窗口 $t+1$ 中解的位置如下所示. 空区域为边界区域, 有

$$x_i^{t+1} = \begin{cases} x_{i+1}^t + \Delta c_i^t, & i = 1; \\ x_{i-1}^t + \Delta c_i^t, & i = k. \end{cases} \quad (13)$$

空区域为中间区域, 有

$$x_i^{t+1} = x_{i-1}^t + F_p \times (x_t^{\text{near}_1} - x_{t-1}^{\text{near}_1}), \quad (14)$$

$$x_i^{t+1} = x_{i+1}^t + F_p \times (x_t^{\text{near}_2} - x_{t-1}^{\text{near}_2}). \quad (15)$$

如式(13)所示, 空区域处于边界区域时, 会出现两种情况: 上边界区域或下边界区域为空. 当上边界区域为空区域, 即 $i = 1$ 时, 通过第 $i+1$ 个区域的个体作为初始个体产生该区域在 $t+1$ 时刻的解. 当下边界区域为空区域, 即 $i = k$ 时, 通过第 $i-1$ 个区域的个体作为初始个体产生该区域在 $t+1$ 时刻的解. 当空区域为中间区域时, 分别通过式(14)和(15)产生第 i 个子区域的解, 并比较两种方式产生解的拥挤距离^[25], 拥挤距离大的解作为第 i 个子区域 $t+1$ 时刻的解.

2.5 算法流程

所提出算法具体流程如下.

step 1: 初始化种群为 P_0 , 并计算种群个体的目标函数值 F_0 .

step 2: 根据2个时刻目标函数值是否有差异检测环境是否发生变化, 若未发生变化, 则转至 step 7, 否则复制当前种群为 P_c .

step 3: 根据式(3)计算环境变化严重程度 $\delta(t)$, 并根据式(4)计算分子区域的个数.

step 4: 初始化1组均匀分布的权重向量 $\lambda(t) = \lambda^1(t), \lambda^2(t), \dots, \lambda^{N(t)}(t)$, 其中权重向量的个数等于子区域的个数, 即 $N(t) = k$.

step 5: 将 P_c 的个体和对应的权重向量执行第2.2节提出的关联操作.

step 6: 对 P_c 的个体执行第2.4节提出的多区域中心点预测策略.

step 7: 采用 MOEA/D-DE 算法^[21] 进行优化, 使用差分进化(DE)运算和多项式变异运算产生新解.

step 8: 进行下一代进化. 若算法达到终止条件, 则返回结果, 算法结束, 否则转至 step 2.

3 实验仿真及结果分析

为了验证所提出算法的有效性, 选择3种性能较好的动态多目标优化算法进行对比测试. 3种算法分别为 PPS 算法^[17]、MOEA/D-KF 算法^[26] 以及 SGEA 算法^[9]. 选取10个动态多目标优化标准测试问题仿真实验, 每种算法均采用 Intel(R) core i5-7500 处理器, 8 GB 内存, 实验环境为 JetBrains PyCharm 2019.1.3 x 64.

3.1 测试函数

选取10个测试函数进行实验仿真. FDA 1~FDA 4 由 Frina 等^[27] 提出的 FDA 测试集得到; Fun 7 是函数 FDA 1 的变体; Fun 8 和 Fun 9 是测试函数 FDA 2 的变体^[24]; DF 2 测试函数来自文献^[28]; dMOP 1 测试函数来自文献^[29]. 本文采用的标准测试函数中, FDA 1、FDA 4 和 DF 2 属于动态问题分类中的第1类问题; FDA 3、Fun 7、Fun 8、Fun 9 和 dMOP 1 属于动态问题分类中的第2类问题; FDA 2、HE 2 属于动态问题分类中的第3类问题.

3.2 参数设置

1) 问题参数设置. 测试函数环境变化的严重程度 $n_t = 10$, 环境变化频率为 $\tau_t = 5, 20$, 决策变量的维度 $n = 10$, 每个算法独立运行 20 次.

2) 种群大小设置为 100. 各算法均以 MOEA/D-DE 为静态优化算法框架, 邻域大小设置为 20, 分解方法为切比雪夫分解法.

3) MCPDMO 算法中步长调节参数 $F_p = 0.4$, 为了确定 F_p 的适当参数, 选取3个测试问题 dMOP 1, FDA 1 和 Fun 7, 并将 F_p 分别设置为 0.2, 0.3, 0.4, 0.5,

表1 MCPDMO算法在 F_p 设置不同值时的MIGD值统计

(τ_t, n_t)	instance	0.2	0.3	0.4
(5,10)	dMOP 1	3.914 7 e-2 (3.279 0 e-2)	3.966 3 e-2 (3.011 1 e-2)	4.224 6 e-2 (5.345 2 e-2)
	FDA 1	1.795 7 e-1 (2.014 9 e-2)	1.883 5 e-1 (1.537 6 e-2)	1.768 1 e-1 (1.701 7 e-2)
	Fun 7	3.166 2 e-2 (2.143 2 e-2)	3.143 7 e-2 (2.456 4 e-2)	3.089 6 e-2 (1.075 0 e-2)
(20,10)	dMOP 1	1.107 9 e-2 (3.279 0 e-2)	1.106 5 e-2 (3.011 1 e-2)	1.119 8 e-2 (5.345 2 e-2)
	FDA 1	1.732 2 e-2 (2.014 9 e-2)	1.744 1 e-2 (1.537 6 e-2)	1.704 6 e-2 (1.701 7 e-2)
	Fun 7	1.441 7 e-2 (2.143 2 e-2)	1.425 8 e-2 (2.456 4 e-2)	1.408 1 e-2 (1.075 0 e-2)
(τ_t, n_t)	instance	0.5	0.6	0.7
(5,10)	dMOP 1	3.952 2 e-2 (3.112 1 e-2)	3.933 6 e-2 (3.078 8 e-2)	3.941 1 e-2 (4.906 6 e-2)
	FDA 1	2.051 4 e-1 (1.046 6 e-2)	1.923 3 e-1 (1.004 8 e-2)	2.014 4 e-1 (1.786 3 e-2)
	Fun 7	3.115 5 e-2 (2.134 4 e-2)	3.232 2 e-2 (2.703 5 e-2)	3.211 1 e-2 (2.132 7 e-2)
(20,10)	dMOP 1	1.117 6 e-2 (3.112 1 e-2)	1.120 1 e-2 (3.078 8 e-2)	1.121 9 e-2 (4.906 6 e-2)
	FDA 1	1.932 1 e-1 (1.046 6 e-2)	1.836 8 e-1 (1.004 8 e-2)	1.941 7 e-1 (1.786 3 e-2)
	Fun 7	1.417 5 e-2 (2.134 4 e-2)	1.423 3 e-2 (2.703 5 e-2)	1.420 9 e-2 (2.132 7 e-2)

0.6, 0.7. 表1给出环境为 $n_t = 10, \tau_t = 5$ 和 $n_t = 10, \tau_t = 20$ 时, MCPDMO算法在 F_p 设置不同值的MIGD值统计. 由表1可见, 当 $F_p = 0.4$ 时, FDA 1 和 Fun 7 均取得了最好的结果. dMOP 1 在环境为 $n_t = 10, \tau_t = 5$ 时, $F_p = 0.2$ 表现较好; 环境为 $n_t = 10, \tau_t = 20$ 时, $F_p = 0.3$ 表现较好. 综合考虑不同测试问题, F_p 设为0.4.

4) 对比算法设置. KF算法中 Q 和 R 对角矩阵的元素值分别为0.04和0.01. PPS算法的AR(p)模型中 $M = 23$, 阶次 p 为3. SGEA算法中交叉概率 $p_c = 1.0$, 其分布指数 $\eta_c = 20$, 变异概率 $p_m = 1/n$, 其分布指数 $\eta_m = 20$.

3.3 评价指标

采用反向代距离 (inverse generation distance, IGD)^[30] 和间距度量 (spacing metric, SP)^[31] 指标评价所提出算法的综合性能, 其中IGD的定义为

$$IGD(P^{t*}, P^t) = \frac{\sum_{v \in P^{t*}} d(v, P^t)}{|P^{t*}|}. \quad (16)$$

其中: t 为当前时刻; P^{t*} 为理想的真实PF上的1组均匀采样点; P^t 为算法在决策空间中求得的Pareto解集; v 为 P^{t*} 中的点; $d(v, P^t) = \min_{u \in P^t} \|F(v) - F(u)\|$ 为 v 与 P^t 之间的距离; $|P^{t*}|$ 为 P^{t*} 的基. IGD作为一个双重指标反映的是真实的Pareto前沿到算法获得近似Pareto前沿的距离, IGD值越小, 表明算法获得的近似Pareto前沿与真实前沿越接近, 算法的收敛性和解的分布度越好. MIGD为算法运行中某些时间段内反向世代距离IGD值的平均值, MIGD定义如下:

$$MIGD(P^{t*}, P^t) = \frac{1}{|T|} \sum_{t \in T} IGD(P^{t*}, P^t). \quad (17)$$

其中: T 为一次运行的离散时间点, $|T|$ 为 T 的基. MIGD值越小表明算法的收敛性和解的分布度越好.

SP指标度量获得 P^t 的分布性, 定义如下:

$$SP = \sqrt{\frac{1}{|P^t - 1|} \sum_{i=1}^{|P^t|} (D_i - \bar{D})^2}. \quad (18)$$

其中: D_i 为 P^t 中第 i 个个体与距其最近解的欧氏距离, \bar{D} 为 D_i 的平均值, D_i 定义为

$$D_i = \min_{u, v \in P^t} \sqrt{\sum_{j=1}^m (f_j(v) - f_j(u))^2}. \quad (19)$$

MSP定义如下:

$$MSP = \frac{\sum_{t \in T} SP(P^t)}{|T|}. \quad (20)$$

与SP一样, MSP值越小, 表明算法获得的解分布越均匀, 算法均匀性越好.

3.4 实验结果及分析

1) IGD性能评价: 4种算法在10个测试函数上分别独立运行20次进行对比实验研究. 其性能评价指标IGD的统计均值和方差如表2所示, 其中加粗部分表示IGD性能指标平均值较小, 表明对应算法具有良好的收敛性和多样性. 由表2可见, 针对环境变化频率 τ_t 取5的情况, 所提出算法在大部分测试函数上的IGD值优于其他3种算法, 这得益于多区域中心点策略对于收敛性及多样性的平衡. SGEA算法在环境变化类型为PF不变的测试函数上表现较好, 得益于该算法在应对环境变化时对之前环境分布性良好解集的重新利用.

针对环境变化频率 τ_t 取20的情况, 所提出算法在大部分测试函数上获得较小的IGD值, MOEAD-KF算法在dMOP 1、FDA 1、Fun 7、Fun 8测试函数上最优. 图2为4种算法在测试函数DF 2、FDA 2、Fun 7、Fun 8、Fun 9、HE 2上的IGD曲线图, 由图2可见, 所提出算法在大部分测试函数上IGD值较小且曲线平稳, 表明MCPDMO算法获得解集更加接近真实的Pareto

表 2 4 种算法的 MIGD 值统计

instance	(τ_t, n_t)	KF	PPS	SGEA	MCPDMO
DF 2	(5,10)	2.181 8 e-1 (1.325 9 e-1)	5.995 2 e-1 (4.037 3 e-1)	1.222 3 e-1 (5.140 0 e-2)	1.465 3 e-1 (9.640 5 e-2)
	(20,10)	2.908 0 e-2 (2.299 7 e-2)	1.231 2 e-1 (1.292 3 e-1)	1.024 7 e-1 (5.120 6 e-2)	2.633 1 e-2 (2.264 6 e-2)
dMOP 1	(5,10)	4.875 2 e-2 (1.380 8 e-1)	1.672 3 e-1 (4.507 0 e-1)	1.044 0 e-2 (2.767 5 e-2)	4.224 6 e-2 (1.295 8 e-1)
	(20,10)	6.173 4 e-3 (1.887 4 e-2)	2.960 8 e-2 (1.871 0 e-1)	1.025 4 e-2 (2.722 7 e-2)	1.119 8 e-2 (1.691 5 e-2)
FDA 1	(5,10)	2.824 0 e-1 (2.438 0 e-1)	1.750 1 e+0 (1.568 9 e+0)	4.800 0 e-2 (3.464 8 e-2)	1.768 1 e-1 (1.213 8 e-1)
	(20,10)	1.134 1 e-2 (4.896 6 e-3)	8.477 4 e-2 (3.849 5 e-1)	2.199 9 e-2 (1.388 3 e-2)	1.704 6 e-2 (4.898 7 e-3)
FDA 2	(5,10)	1.898 2 e-1 (1.736 9 e-1)	4.205 3 e-1 (3.705 2 e-1)	2.652 0 e-2 (1.206 4 e-2)	1.113 3 e-2 (1.029 7 e-2)
	(20,10)	1.835 9 e-1 (1.552 5 e-1)	1.889 7 e-1 (1.620 0 e-1)	1.647 8 e-2 (1.284 0 e-2)	5.632 6 e-3 (1.148 2 e-3)
FDA 3	(5,10)	4.566 2 e-1 (1.777 3 e-1)	1.642 1 e+0 (1.482 6 e+0)	8.542 3 e-2 (6.508 6 e-2)	1.125 9 e-1 (8.128 1 e-2)
	(20,10)	2.925 8 e-1 (1.524 6 e-1)	3.807 4 e-1 (2.222 0 e-1)	5.330 4 e-2 (5.657 9 e-2)	6.986 1 e-2 (6.619 8 e-2)
FDA 4	(5,10)	1.509 5 e-1 (8.075 3 e-2)	4.027 9 e-1 (2.719 5 e-1)	8.443 0 e-2 (4.399 6 e-2)	2.810 2 e+0 (3.191 7 e+0)
	(20,10)	5.438 3 e-2 (5.220 9 e-3)	8.917 0 e-2 (4.601 7 e-2)	5.187 6 e-2 (1.972 3 e-2)	2.454 5 e+0 (3.104 9 e+0)
Fun 7	(5,10)	3.214 9 e-2 (3.534 1 e-2)	1.089 7 e-1 (9.712 2 e-2)	2.302 4 e-1 (2.789 0 e-1)	3.089 6 e-2 (3.491 1 e-2)
	(20,10)	8.539 6 e-3 (5.227 2 e-2)	1.887 4 e-2 (4.714 0 e-2)	3.346 2 e-2 (2.334 2 e-2)	1.408 1 e-2 (1.400 5 e-2)
Fun 8	(5,10)	1.976 4 e-2 (1.535 3 e-2)	1.714 3 e-1 (5.946 4 e-2)	3.937 0 e-2 (2.409 7 e-2)	1.656 1 e-2 (9.747 5 e-3)
	(20,10)	5.682 4 e-3 (2.352 2 e-3)	1.730 4 e-2 (1.763 7 e-2)	3.351 0 e-2 (1.939 9 e-2)	7.660 5 e-3 (2.164 4 e-3)
Fun 9	(5,10)	7.971 1 e-2 (1.106 7 e-1)	4.343 7 e-1 (1.716 2 e-1)	9.299 3 e-2 (1.212 5 e-1)	7.549 0 e-2 (1.066 8 e-1)
	(20,10)	4.491 8 e-2 (1.003 6 e-1)	5.688 1 e-2 (1.168 3 e-1)	7.838 0 e-2 (1.134 1 e-1)	4.427 8 e-2 (1.003 5 e-1)
HE 2	(5,10)	8.015 3 e-1 (4.456 3 e-1)	7.831 5 e-1 (5.051 9 e-1)	2.425 5 e-1 (8.125 1 e-2)	2.342 6 e-1 (2.519 7 e-1)
	(20,10)	5.403 2 e-1 (4.435 3 e-1)	1.644 6 e+0 (2.986 2 e-1)	2.275 0 e-1 (7.817 5 e-2)	1.542 4 e-1 (8.559 0 e-2)

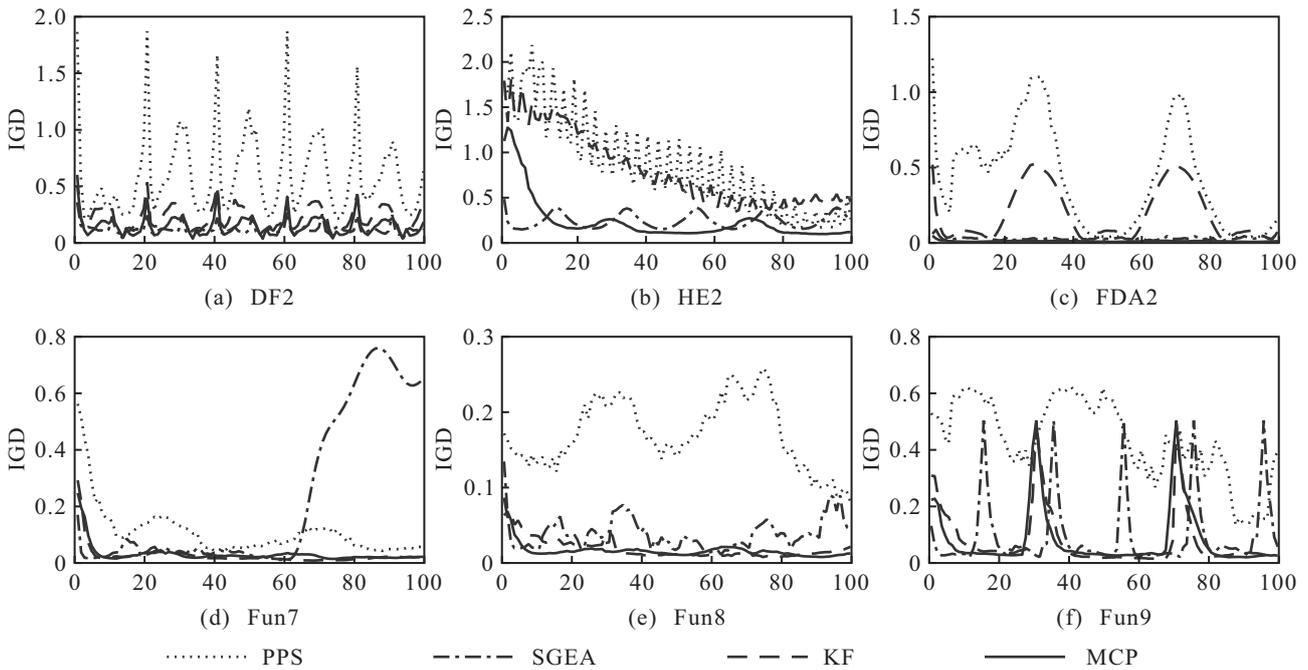


图 2 各算法在各测试函数的 IGD 对比, 环境为 $n_t = 10, \tau_t = 5$

前沿, 且算法更加稳定. 这主要得益于 MCPDMO 算法采取的多中心点预测策略, 可以充分描述不同类型前沿或解集的变化.

2) SP 性能评价: 4 种算法在 10 个测试函数上分别独立运行 20 次进行对比实验研究. 其性能评价指标 SP 的统计均值和方差如表 3 所示, 以比较算法在各个环境下的分布性. 由表 3 可见, 针对环境变化频率 τ_t 取 5 的情况, 所提出算法在 10 个测试函数中 7 个测试函数上的 SP 值优于其他 3 种算法, 这主要是由于不同子区域的个体独立进化保证了种群的分布性, 但在

测试函数 FDA 1 和 FDA 4 上略差于 SGEA 算法. 针对环境变化频率 τ_t 取 20 的情况, 所提出算法在 10 个测试函数中 5 个测试函数上的 SP 值优于其他 3 种算法, 而 SGEA 算法在 FDA 4 上最优, MOEA/D-KF 算法在 dMOP 1、FDA 1、FDA 3、Fun 7 上最优. 尤其当环境变化频率 τ_t 取 5 时, MCPDMO 算法具有明显的优势, 表明所提出算法能够更好地保持种群的多样性和分布性, 且更能适应剧烈变化的环境. MCPDMO 在分解的基础上划分不同的子区域, 保证了算法的分布性.

表3 4种算法的MSP值统计

instance	(τ_t, n_t)	KF	PPS	SGEA	MCPDMO
DF2	(5,10)	2.3318e-1 (1.5230e-1)	2.9718e-1 (1.7610e-1)	2.3310e-1 (1.6182e-1)	4.9607e-2 (3.0865e-2)
	(20,10)	2.9080e-2 (2.2997e-2)	1.1031e-1 (9.8783e-2)	2.8081e-1 (1.4410e-1)	1.1875e-2 (1.0390e-2)
dMOP1	(5,10)	2.5060e-2 (7.0224e-1)	3.7895e-2 (8.6996e-2)	2.7720e-2 (3.4591e-2)	1.7001e-2 (3.9099e-2)
	(20,10)	3.4319e-3 (1.2284e-2)	8.4297e-3 (4.0507e-2)	2.1883e-2 (3.2270e-2)	6.1584e-3 (1.1386e-2)
FDA1	(5,10)	2.1476e-1 (1.7679e-1)	4.2285e-1 (1.9702e-1)	6.0437e-2 (4.6166e-2)	1.2442e-1 (7.9900e-2)
	(20,10)	1.4620e-2 (5.7877e-3)	3.9795e-2 (8.1678e-2)	4.3266e-2 (4.1184e-2)	1.9974e-2 (9.6665e-3)
FDA2	(5,10)	8.8785e-2 (8.5077e-2)	8.1040e-2 (5.7050e-2)	6.1682e-2 (4.8975e-2)	2.0302e-2 (1.8808e-2)
	(20,10)	9.4284e-2 (8.9082e-2)	1.8897e-2 (8.8286e-2)	4.9662e-2 (6.2034e-2)	9.2640e-3 (2.4622e-3)
FDA3	(5,10)	1.2324e-1 (9.0563e-2)	2.3262e-1 (1.6959e-1)	1.6047e-1 (1.6009e-1)	1.0017e-1 (4.7178e-2)
	(20,10)	7.4767e-2 (4.5598e-2)	9.1410e-2 (5.2344e-2)	1.4399e-1 (1.5429e-1)	9.0182e-2 (5.2811e-2)
FDA4	(5,10)	1.2690e-1 (6.2474e-2)	1.8911e-1 (8.2259e-2)	7.0745e-2 (3.7730e-2)	3.8630e-1 (1.2644e-1)
	(20,10)	5.1890e-2 (4.0837e-3)	6.1169e-2 (3.1001e-2)	4.5624e-2 (1.0096e-2)	3.8392e-1 (1.0232e-1)
Fun7	(5,10)	8.8356e-2 (1.1208e-2)	2.8148e-1 (2.5041e-1)	3.9195e-1 (5.0449e-1)	8.3429e-2 (1.0428e-1)
	(20,10)	2.6440e-2 (1.5739e-2)	4.6017e-2 (1.2116e-2)	9.7887e-2 (7.3113e-2)	5.0989e-2 (4.8313e-2)
Fun8	(5,10)	5.5930e-1 (6.2599e-2)	4.8955e-1 (1.7930e-1)	8.2375e-2 (6.9657e-2)	1.1813e-2 (1.4244e-2)
	(20,10)	1.1156e-2 (9.7102e-3)	4.7882e-2 (6.9483e-2)	7.9614e-2 (6.6269e-2)	3.9707e-3 (2.0811e-3)
Fun9	(5,10)	2.0326e-1 (2.7891e-1)	9.9159e-1 (3.1361e-1)	2.0052e-1 (2.6565e-1)	1.9367e-1 (2.6956e-1)
	(20,10)	1.1559e-1 (2.4018e-1)	1.5023e-1 (2.8387e-1)	1.9489e-1 (2.6962e-1)	1.1286e-1 (2.3805e-1)
HE2	(5,10)	1.3539e+0 (3.2282e-1)	1.1376e+0 (3.4218e-1)	6.2289e-1 (3.1332e-1)	5.1640e-1 (2.5575e-1)
	(20,10)	1.1204e+0 (4.0801e-1)	1.4945e+0 (1.0496e-1)	7.4272e-1 (1.4192e-1)	5.4410e-1 (1.7378e-1)

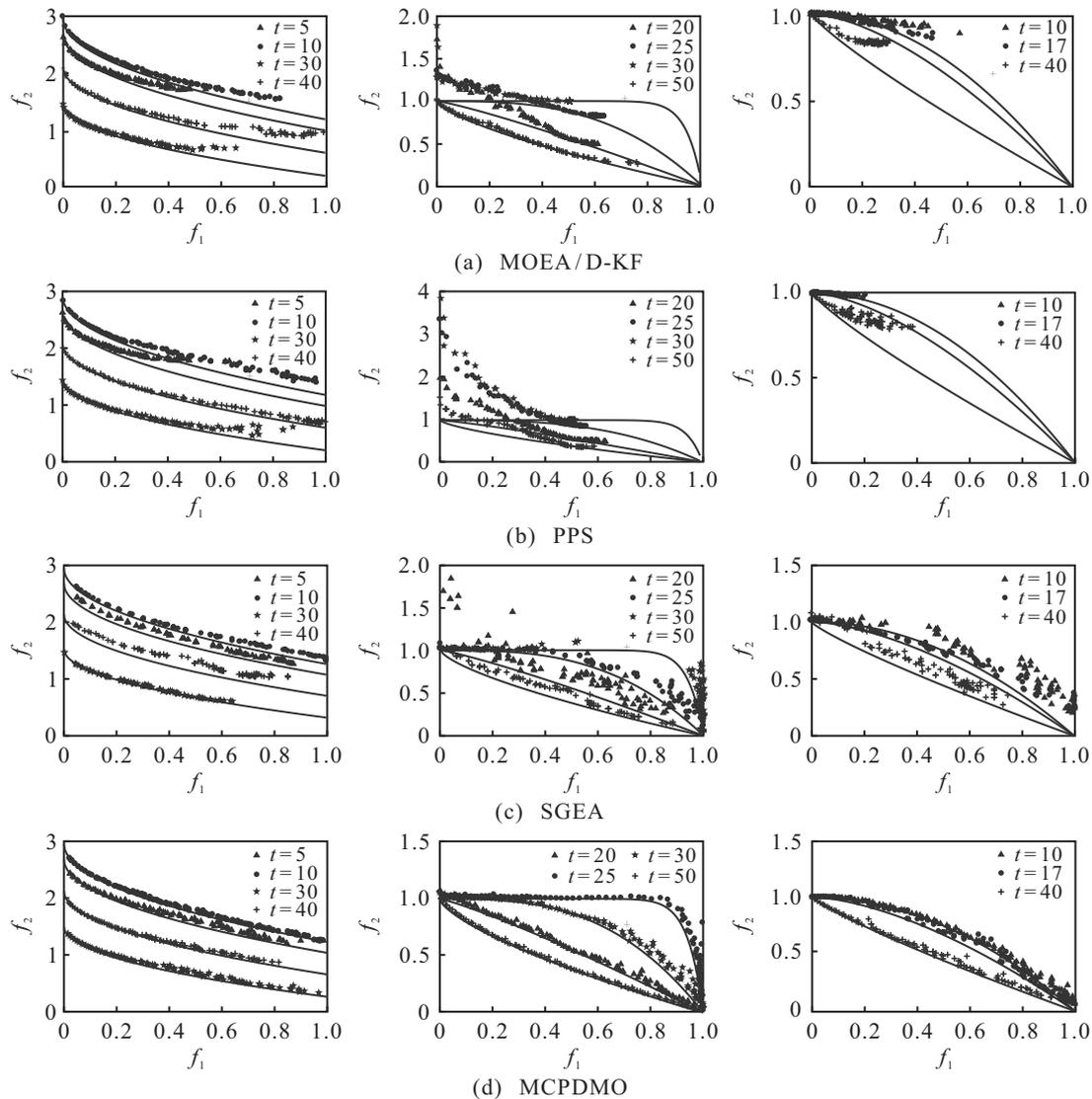


图3 各算法在各测试函数不同时刻t的前沿分布,环境为 $n_t=10, \tau_t=5$

由IGD和SP性能指标分析表明,所提出算法较其他算法具有优越性.为了观察不同时刻4种算法得到的解的分布情况,选取测试函数Fun 7在 t 为5、10、30、40时4种算法得到的解,结果如图3所示.由图3可见,4种算法随着时间的累积,效果逐渐变好,但是MOEA/D-KF和PPS的分布性依然较差.SGEA算法虽然获得的解比较均匀,但是收敛性较差.而对于MCPDMO而言,在前期就已经取得了较好的效果.选取测试函数Fun 8在 t 为20、25、30、50时刻解的分布,结果如图3所示.由图3可见,其他3种算法均不能很好地收敛到真实前沿,而MCPDMO在保证收敛性的同时解集的分布也很均匀.同时,选取测试函数Fun 9在 t 为10、17、40时刻解的分布,结果如图3所示.图中线为不同时刻真实前沿面,点为不同时刻求得的解集.由图3可见,MCPDMO算法更接近真实解且分布更均匀.

综上所述:所提出算法通过分解的方法将解分配至不同子区域,通过同一子区域中心点建立时间序列,当动态多目标优化问题的环境变化时,不同子区域的时间序列同时对该环境下的最优解进行预测;同时针对无关联个体的区域提出了有效措施.分解的方法可以使解集分布更加均匀,同时预测多个区域的中心点使预测精度更高,使得所提出算法在加快收敛速度的同时保持多样性和分布性.通过与MOEA/D/KF、PPS、SGEA算法的实验结果比较可得,MCPDMO算法能够更好地适应动态环境的变化.

4 结论

对于动态多目标进化算法而言,当环境发生变化时,如何及时追踪变化的PF或PS并保持良好的分布性是其核心问题.本文提出了一种基于多区域中心点的动态多目标优化算法,该算法根据环境变化严重程度将个体划分到不同子区域,并对不同子区域的中心点建立时间序列,利用一阶差分模型对时间序列进行预测,具有较高的可适应性和良好的分布性.

利用动态多目标优化标准测试问题与3种算法在不同环境下进行性能测试,结果表明所提出算法能够快速追踪变化的PF且分布性较好.但是,对于3目标的测试问题,收敛性有待进一步改善.此外,实际生活中的问题都较为复杂且规律性不强,对于预测而言仍具有很大挑战,是未来研究的重点.

参考文献(References)

[1] Hu Z Y, Wei Z H, Sun H, et al. Optimization of metal rolling control using soft computing approaches: A review[J]. Archives of Computational Methods in

- Engineering, 2021, 28(2): 405-421.
- [2] 杨景明, 侯新培, 崔慧慧, 等. 基于融合多策略改进的多目标粒子群优化算法[J]. 控制与决策, 2018, 33(2): 226-234.
(Yang J M, Hou X P, Cui H H, et al. Improved multi-objective particle swarm optimization algorithm based on integrating multiply strategies[J]. Control and Decision, 2018, 33(2): 226-234.)
- [3] 刘淳安. 动态多目标优化进化算法研究综述[J]. 海南大学学报: 自然科学版, 2010, 28(2): 176-182.
(Liu C A. Research on dynamic multiobjective optimization evolutionary algorithms[J]. Natural Science Journal of Hainan University, 2010, 28(2): 176-182.)
- [4] Ma X M, Yang J M, Sun H, et al. Multiregional co-evolutionary algorithm for dynamic multiobjective optimization[J]. Information Sciences, 2021, 545: 1-24.
- [5] Peng Z, Zheng J H, Zou J. A population diversity maintaining strategy based on dynamic environment evolutionary model for dynamic multiobjective optimization[C]. IEEE Congress on Evolutionary Computation. Piscataway: IEEE, 2014: 274-281.
- [6] Zou J, Li Q Y, Yang S X, et al. A dynamic multiobjective evolutionary algorithm based on a dynamic evolutionary environment model[J]. Swarm and Evolutionary Computation, 2019, 44: 247-259.
- [7] Akbari R, Hedayatzadeh R, Ziarati K, et al. A multi-objective artificial bee colony algorithm[J]. Swarm and Evolutionary Computation, 2012, 2: 39-52.
- [8] Liu R C, Li J X, fan J, et al. A coevolutionary technique based on multi-swarm particle swarm optimization for dynamic multi-objective optimization[J]. European Journal of Operational Research, 2017, 261(3): 1028-1051.
- [9] Jiang S Y, Yang S X. A steady-state and generational evolutionary algorithm for dynamic multiobjective optimization[J]. IEEE Transactions on Evolutionary Computation, 2017, 21(1): 65-82.
- [10] Ruan G, Yu G, Zheng J H, et al. The effect of diversity maintenance on prediction in dynamic multi-objective optimization[J]. Applied Soft Computing, 2017, 58: 631-647.
- [11] Nasiri B, Meybodi M, Ebadzadeh M. History-Driven Particle Swarm Optimization in dynamic and uncertain environments[J]. Neurocomputing, 2016, 172: 356-370.
- [12] Peng Z, Zheng J H, Zou J, et al. Novel prediction and memory strategies for dynamic multiobjective optimization[J]. Soft Computing, 2015, 19(9): 2633-2653.
- [13] Azzouz R, Bechikh S, Said L B. A dynamic multi-objective evolutionary algorithm using a change

- severity-based adaptive population management strategy[J]. *Soft Computing*, 2017, 21(4): 885-906.
- [14] Liu R C, Chen Y Y, Ma W P, et al. A novel cooperative coevolutionary dynamic multi-objective optimization algorithm using a new predictive model[J]. *Soft Computing*, 2014, 18(10): 1913-1929.
- [15] Jiang M, Huang Z Q, Qiu L M, et al. Transfer learning-based dynamic multiobjective optimization algorithms[J]. *IEEE Transactions on Evolutionary Computation*, 2018, 22(4): 501-514.
- [16] Wu Y, Jin Y C, Liu X X. A directed search strategy for evolutionary dynamic multiobjective optimization[J]. *Soft Computing*, 2015, 19(11): 3221-3235.
- [17] Zhou A M, Jin Y C, Zhang Q F. A population prediction strategy for evolutionary dynamic multiobjective optimization[J]. *IEEE Transactions on Cybernetics*, 2014, 44(1): 40-53.
- [18] Zou J, Li Q Y, Yang S X, et al. A prediction strategy based on center points and knee points for evolutionary dynamic multi-objective optimization[J]. *Applied Soft Computing*, 2017, 61: 806-818.
- [19] Ma X M, Yang J M, Sun H, et al. Feature information prediction algorithm for dynamic multi-objective optimization problems[J]. *European Journal of Operational Research*, 2021, 295(3): 965-981.
- [20] Cao L L, Xu L H, Goodman E D, et al. Evolutionary dynamic multiobjective optimization assisted by a support vector regression predictor[J]. *IEEE Transactions on Evolutionary Computation*, 2020, 24(2): 305-319.
- [21] 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.
- [22] Hu Z Y, Yang J M, Cui H H, et al. MOEA 3D: A MOEA based on dominance and decomposition with probability distribution model[J]. *Soft Computing*, 2019, 23(4): 1219-1237.
- [23] Hu Z Y, Yang J M, Sun H, et al. An improved multi-objective evolutionary algorithm based on environmental and history information[J]. *Neurocomputing*, 2017, 222: 170-182.
- [24] Rong M, Gong D W, Zhang Y, et al. Multidirectional prediction approach for dynamic multiobjective optimization problems[J]. *IEEE Transactions on Cybernetics*, 2019, 49(9): 3362-3374.
- [25] Deb K, Pratap A, Agarwal S, et al. A fast and elitist multiobjective genetic algorithm: NSGA-II[J]. *IEEE Transactions on Evolutionary Computation*, 2002, 6(2): 182-197.
- [26] Muruganantham A, Tan K C, Vadakkepat P. Evolutionary dynamic multiobjective optimization via Kalman filter prediction[J]. *IEEE Transactions on Cybernetics*, 2016, 46(12): 2862-2873.
- [27] Farina M, Deb K, Amato P. Dynamic multiobjective optimization problems: Test cases, approximation, and applications[C]. *Evolutionary Multi-Criterion Optimization*. Berlin, Heidelberg: Springer-Verlag, 2003: 311-326.
- [28] Biswas S, Das S, Suganthan P N, et al. Evolutionary multiobjective optimization in dynamic environments: A set of novel benchmark functions[C]. *2014 IEEE Congress on Evolutionary Computation*. Piscataway: IEEE, 2014: 3192-3199.
- [29] Goh C K, Tan K C. A competitive-cooperative coevolutionary paradigm for dynamic multiobjective optimization[J]. *IEEE Transactions on Evolutionary Computation*, 2009, 13(1): 103-127.
- [30] Zhou A M, Jin Y C, Zhang Q F. A population prediction strategy for evolutionary dynamic multiobjective optimization[J]. *IEEE Transactions on Cybernetics*, 2014, 44(1): 40-53.
- [31] Deb K, Pratap A, Agarwal S, et al. A fast and elitist multiobjective genetic algorithm: NSGA-II[J]. *IEEE Transactions on Evolutionary Computation*, 2002, 6(2): 182-197.

作者简介

马学敏(1994—),女,博士生,从事动态多目标优化的研究, E-mail: mxm0217@163.com;

杨景明(1957—),男,教授,博士生导师,从事轧制过程建模、人工智能与神经网络应用、轧制过程自动控制等研究, E-mail: yangjm@ysu.edu.cn;

孙浩(1985—),男,讲师,博士,从事智能优化算法、深度神经网络、轧制过程控制等研究, E-mail: sunhao@ysu.edu.cn;

呼子宇(1989—),男,讲师,博士,从事复杂系统建模、多目标进化算法、人工智能等研究, E-mail: hzy@ysu.edu.cn;

韦钦楠(1997—),男,硕士生,从事多任务优化的研究, E-mail: sciencewqn@163.com.

(责任编辑:魏冰)