

基于多邻居结构的自适应元胞差分算法

王亚良,倪晨迪,金寿松

(浙江工业大学机械工程学院,浙江杭州 310023)

摘 要: 针对传统多目标优化算法在求解 Pareto 解集时存在全局搜索能力与局部寻优能力无法得到有效平衡的问题,提出了一种基于多邻居结构的自适应元胞差分算法. 该算法在保留传统元胞差分算法进化特点的基础上,使用更加丰富的多邻居结构替换原有的单一邻居结构,并且依据相应元胞个体的性能优劣来对其邻居结构进行选择分配. 同时,面对进化过程中的复杂性能需求,算法定义了一种周期性变化的变异策略来实现不同进化阶段的自适应调节. 最后,利用 DTLZ 系列测试函数对算法性能进行测试,并通过与四种经典的多目标优化算法相比较,证明了改进后的算法拥有更好的收敛性与分布性.

关键词: 多目标优化; 多邻居结构; 随机扰动; 自适应变异; 元胞自动机; 进化策略

中图分类号: TP18 **文献标识码:** A **文章编号:** 0372-2112 (2021)03-0578-08

电子学报 URL: <http://www.ejournal.org.cn> **DOI:** 10.12263/DZXB.20200228

Adaptive Cellular Differential Evolutionary Algorithm Based on Multi-neighborhood Structure

WANG Ya-liang, NI Chen-di, JIN Shou-song

(College of Mechanical Engineering, Zhejiang University of Technology, Hangzhou, Zhejiang 310023, China)

Abstract: To solve the problem that the global search ability and local search ability of traditional multi-objective evolutionary algorithm cannot be effectively balanced when solving the Pareto solution set, an adaptive cellular differential evolutionary algorithm based on multi-neighborhood structure is proposed. Based on the characteristics of the traditional cellular differential evolutionary algorithm, the improved algorithm uses a richer multi-neighbor structure to replace the original single neighbor structure, and the neighbor structure is adjusted reasonably according to the performance of the corresponding individual. At the same time, in the face of the complex requirements in the whole evolution process, the algorithm defines a mutation strategy with periodic variation to realize the adaptive adjustment in different evolution stages. Finally, the DTLZ series of test functions are used to test the performance of the algorithm. Compared with four classical multi-objective optimization algorithms, it is proved that the improved algorithm has better convergence performance and diversity of solution set.

Key words: multi-objective optimization; multi-neighborhood structure; random disturbance; adaptive mutation; cellular automata; evolutionary strategy

1 引言

在工程实践中,多目标优化问题(Multi-objective Optimization Problems, MOPs)在许多方面都有着重要的应用,例如:机械优化设计、车间布局调度、物流运输规划等.与单目标优化问题有所不同,多目标优化问题需要同时对多个子目标进行优化,但是决策变量上的相

互制约,使其无法通过全局最优解来实现问题的求解.因此决策者只能依据需求对其进行折衷处理,得到一个实现各子目标相对优化的非劣解集,而该解集也被称之为 Pareto 解集.

目前,为了实现对 Pareto 解集的求解,多目标优化算法是一种得到广泛应用的方法.近年来,许多优秀的多目标优化算法得到不断的发展,其中典型的算法有:

非支配排序多目标遗传算法(Non-dominated Sorting Genetic Algorithm II, NSGA-II)^[1]、改进的强度 Pareto 算法(Strength Pareto Evolutionary Algorithm2, SPEA2)^[2]和多目标粒子群算法(Multi-Objective Particle Swarm Optimization algorithm, MOPSO)^[3]等.

2008 年, Durillo 通过将元胞化的种群结构与差分进化方式相结合, 并以此提出了元胞差分算法^[4](Cellular Differential Evolution, CellIDE). 此后, 许多学者为了提升该算法的性能, 对其进行了更加深入的研究并提出了相应的改进方案. 张屹等^[5]在进化操作中采用定向差分的方式来加强算法的收敛性, 并利用带扰动的多项式变异实现算法多样性的维护. 王亚良等^[6]将种群的进化过程划分为两个阶段, 在不同的阶段分别采用不同的邻居结构形式和外部种群维护方式, 平衡了算法的全局搜索和局部寻优能力. 詹腾等^[7]将多种变异策略引入算法中, 并结合改进的性能评估方式实现了算法多样性的提高. 王福才等^[8]为了提高算法的收敛效率, 将外部种群与当前进化种群相融合, 利用非支配排序的方式从混合种群中筛选出精英个体来组成下一代的父本种群, 使得外部种群中的精英个体直接有效的参与到进化过程中.

在元胞差分算法的应用过程中, 算法在不同的改进方式下都取得了不同程度的优化效果. 但是在设计过程中, 却少有学者将元胞体的性能特点与进化过程的周期特性进行充分的考虑分析. 针对上述问题, 本文提出了一种基于多邻居结构的自适应元胞差分算法(Adaptive Cellular Differential Evolutionary algorithm based on Multi-neighborhood Structure, MS-ACellIDE), 该算法通过调整种群中个体的邻居结构来增强算法的搜索能力, 避免其出现陷入局部最优的情况. 并且, 利用两种不同的差分策略来满足不同进化阶段的性能需求, 使算法的优化效果得到提高.

2 MS-ACellIDE 算法

元胞差分算法是一种结合了元胞自动机(Cellular Automata, CA)结构特点与差分算法(Differential Evolution, DE)进化机制所形成的耦合算法. 不同于常规种群的进化模式, 元胞差分算法利用元胞为载体来实现种群的不断进化.

元胞差分算法的进化过程如图 1 所示: 首先, 需要将种群中的个体分布于二维拓扑网格中. 在进化过程中, 当前个体需要与邻居结构内的元胞个体来共同组成父本种群, 父本种群通过变异、交叉的进化操作生成子代个体. 如果生成的子代个体在性能上优于当前个体, 则该子代个体替换当前个体进入进化种群. 与此同时, 该子代个体还应复制进入算法所设置的外部种群.

在每一代种群进化结束后, 储存在外部种群中的非支配个体利用随机反馈的方式替换部分原始进化种群的个体来实现种群性能的提高.

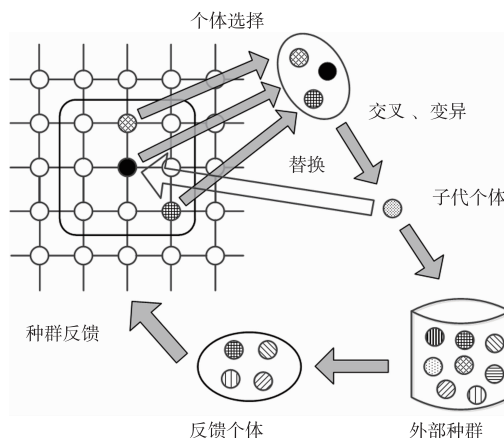


图1 元胞差分算法的进化过程示意图

2.1 邻居结构自适应策略

作为一种多目标优化算法, 元胞差分算法采用元胞化的方式实现了种群个体的结构化分布. 根据算法的进化规则, 元胞个体的进化过程需要在设定的邻居结构内进行. 所以对于元胞个体而言, 参与进化的邻居个体及邻居结构对于算法的优化效果有着重要的影响.

在元胞差分算法中, 元胞个体通常采用固定的 C9 型(Moore 型)邻居结构来实现进化的过程^[9]. 在该邻居结构下, 算法拥有良好的全局搜索能力, 但是在一定程度上会影响算法的收敛效率. 并且由于邻居个体数量的限制, 算法后期容易因为个体不断趋近于最优解而导致扰动变量的数值减小, 进而在种群中出现“早熟”的现象. 尤其对于算法种群中性能表现较为优异的精英个体, 受到邻域范围的限制, 其实现持续进化的难度较大. 而一旦种群中的部分个体因为此原因无法实现对 Pareto 最优前端的收敛, 那么其将直接导致算法优化效果的下降. 并且, 由于结构内个体之间的相互渗透影响, 种群整体的表现性能也会受到影响. 为了避免上述问题的发生, 元胞差分算法可以依据相应元胞个体的性能特点来对其邻居结构进行调整, 进而实现算法优化能力的提升^[10].

为了实现对邻居结构的调节, 本文在保留原始 Moore 型邻居结构的基础上引入了 C13 与 C25 两种邻居结构来丰富种群的结构类型^[10]. 同时, 为了让多种邻居结构得到合理的应用, 本文对邻居结构的分配过程进行了设计, 如图 2 所示: 当进化过程开始时, 算法利用个体的支配等级与拥挤距离对种群个体进行优劣性能的排序. 每当个体进行邻居结构的选择时, 算法需要通过判断个体的隶属类型来进行对应邻居结构的分配. 如果当前个体为优

势个体, 为了加强其与相邻元胞个体的联系且避免过早收敛的问题, 该类型的个体需要算法分配给予领域范围较广的结构, 此时应采用 C25 型邻居结构; 如果当前个体隶属于常规个体时, 其进化需求相较于前者会有所下降, 那么此时邻居结构可以采用 C13 型结构来实现个体的进化过程; 而当前个体属于劣势个体时, 其可实现进化的空间相较于其他两种类型较大, 故利用 Moore 型结构来提高该类型个体的优化效果. 当不同性能特点的个体在不同类型的邻居结构下实现进化, 该方式使得种群中的个体都可以依据自己的性能表现得到合理的进化, 并让算法的性能得到提高.

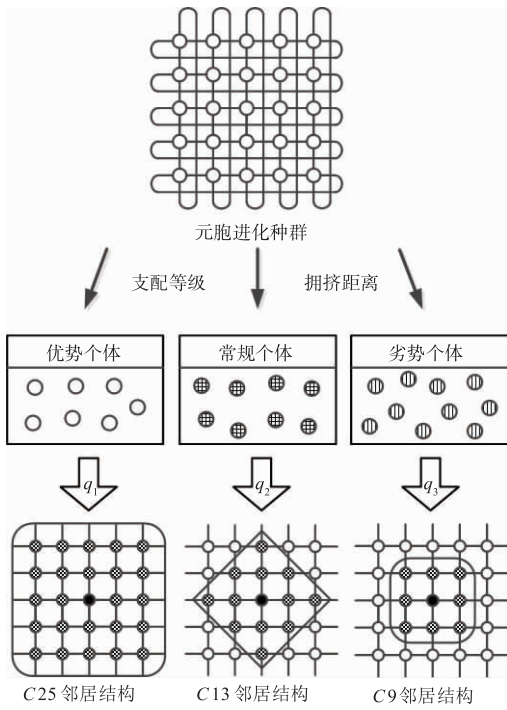


图2 邻居结构变化过程示意图

在图2中, q_1 、 q_2 、 q_3 为对应优势个体、常规个体和劣势个体在种群所占的比例. 在个体隶属类型的判断过程中, q_1 、 q_2 、 q_3 的数值大小扮演了重要的角色, 并且直接影响了实际的进化过程. 在相关的论文中^[9,11], q_1 、 q_2 、 q_3 的数值大多被设置为固定的数值. 但是固定的种群个体数量的设计无法满足种群在进化过程中的复杂需求, 所以为了算法的性能可以得到进一步的提高, 本文对该组数值的大小进行了动态的设计, 如式(1)所示:

$$\begin{cases} q_1 = q_{l1} + \frac{1}{1 + (\frac{G_{\max} - G}{G_{\max}})} (q_{u1} - q_{l1}) \\ q_2 = q_{l2} + \frac{1}{1 + (\frac{G_{\max} - G}{G_{\max}})} (q_{u2} - q_{l2}) \\ q_3 = 1 - q_1 - q_2 \end{cases} \quad (1)$$

q_{l1} 与 q_{u1} 为优势种群在进化过程中所设定的最小与最大种群比例; 同理, q_{l2} 与 q_{u2} 为种群中常规种群最小和最大种群比例; G 代表当前进化代数, G_{\max} 代表最大进化代数. 其中, 为了保证种群在进化前期能够充分发挥原始邻居结构寻优能力强的特点, q_{l1} 、 q_{l2} 取值为 0.1; 同时, 为了使得多邻居结构的优势得到发挥, q_{u1} 、 q_{u2} 取值为 0.3. 当该分配策略得到应用时, 可以使算法在进化前期注重全局搜索, 后期保证种群中的个体得到充分进化, 让种群的全局搜索能力与局部寻优能力实现有效的平衡.

2.2 周期自适应的差分变异机制

对于元胞差分算法而言, 针对元胞化结构进化效率不高的缺点, 算法依托差分进化的方式来改善种群中个体的进化问题. 而作为该过程中的重要环节, 差分变异机制对于算法的最终优化效果有着重要的引导作用^[12-14]. 尤其对于不同周期内种群的性能需求, 差分变异机制可以通过阶段化的调节来实现满足. 因此, 本文依据相应的变异策略与种群进化周期的特点, 采用了一种周期自适应的变异算子:

$$V_{i,G} = \begin{cases} X_{i,G} + F_1(X_{\text{rand},G} - X_{i,G}) + F_2(X_{r1,G} - D_{r2,G}), & \text{if rand} < \tau_1 \\ X_{i,G} + F_1(X_{\text{best},G} - X_{i,G}) + F_2(X_{r1,G} - D_{r2,G}), & \text{otherwise} \end{cases} \quad (2)$$

$$\tau_1 = (\lambda_{\max} - \lambda_{\min}) \cdot \sin\left(\frac{\pi}{2} \cdot \frac{G_{\max} - G}{G_{\max}}\right) + \lambda_{\min} \quad (3)$$

其中, $X_{\text{rand},G}$ 、 $X_{r1,G}$ 是从当前邻居范围内随机选取的个体; $D_{r2,G}$ 为算法设定的扰动个体; $X_{\text{best},G}$ 为邻居结构中的最优个体; $X_{i,G}$ 为基准个体; F_1 和 F_2 对应差分向量的缩放因子; λ_{\max} 与 λ_{\min} 为算法变异机制的最大和最小概率; τ_1 为设置的时间阈值.

$$d_{r2,G}^j = \begin{cases} L^j + \text{rand} \cdot (U^j - L^j), & \text{if rand} < \tau_2 \\ x_{r2,G}^j, & \text{otherwise} \end{cases} \quad (4)$$

$$\tau_2 = \frac{1}{100} (1 + 9 \cdot 10^{5(t-1)}) \quad (5)$$

其中, $d_{r2,G}^j$ 为 $D_{r2,G}$ 中的第 j 个个体; $x_{r2,G}^j$ 是邻居结构内随机个体 $X_{r2,G}$ 中的第 j 个个体; L^j 和 U^j 相应元胞个体变量值的上下边界值; τ_2 为设置的时间阈值; $t = G/G_{\max}$, 即当前进化代数除以最大进化代数所得数值.

通过该变异机制, 算法在不同进化阶段实现了对收敛性与多样性的维护. 根据周期函数的设定, 在算法的早期阶段, 进化种群会较多地采用 DE/current-to-rand/1/bin 的变异机制. 在该情况下, 全局搜索的能力得到强调, 个体多样性得到提高. 而随着算法进程的不断深入, DE/current-to-best/1/bin 变异机制被选择的概率随着阈值的降低而不断提高. 依据该变异策略, 算法的收敛性得到增

强,使收敛不佳的问题得到避免.在差分算子的设计过程中,本文还引入了一个随机扰动向量 $d_{n,G}^i$,该扰动向量的应用使得算法在进化阶段的后期仍然可以通过随机扰动的方式来避免算法陷入局部最优的问题.

2.3 MS-ACellDE 算法流程

依据上述章节中对算法的改进措施,本文所提出的 MS-ACellDE 算法流程如下:

MS-ACellDE 算法

- 输入: 种群规模,最大迭代次数,外部种群规模,交叉概率,缩放因子,优势种群的最大、最小种群比例,常规种群的最大、最小种群比例
- 输出: 最后一代外部种群中所有的解
- Step1** 将随机生成的元胞个体分布在二维环形拓扑结构中来形成初始种群.同时,建立外部种群并复制初始种群中的个体进入外部种群;
- Step2** 据种群中个体的性能特点,算法通过设定的分配机制赋予个体对应的邻居结构.在邻居结构范围内,当前个体与相应的邻居个体共同通过差分进化的方式生成子代个体.在该过程中,对于不同进化阶段的个体,算法采用两种不同性能特点的变异机制来实现子代个体的生成;
- Step3** 将生成的子代个体与当前个体进行比较,若子代个体成功实现对当前个体的支配,则该子代个体进入原始种群替换对应的当前元胞个体,并进入外部种群;
- Step4** 在每一代进化周期结束之后,利用个体支配等级和拥挤距离对外部种群数量进行修剪,结束后将该种群中的非劣个体部分反馈分配到原始种群中;
- Step5** 判断算法进程是否满足终止条件,如满足,则终止算法进程输出最终所获非支配解集;如果不满足,则循环执行步骤 2, 3, 4.

3 算法性能测试及其结果比较分析

为了明确基于多邻居结构的自适应元胞差分算法的性能情况,需要对其进行全面准确的性能评估.因此,本文采用 DTLZ 系列函数^[15]作为测试基准函数来对算法性能进行测试,并采用三种不同的性能评价指标对测试得到的计算结果进行评价.

3.1 性能评价指标

(1) 世代距离 (Generational Distance, GD)

世代距离是用来表达算法计算所得 Pareto 前端收敛到真实最优 Pareto 前端的程度.

$$GD = \frac{1}{n} \sqrt{\sum_{i=1}^n d_i^2} \quad (6)$$

式中, n 为覆盖 Pareto 面的种群采样点数量, d_i 为第 i 个解的目标函数所构成的向量与最优 Pareto 前端之间的最近距离. GD 值越小,表明算法收敛能力越强.

(2) 分布指标 (Spacing, S)

分布指标 S 通过计算每个解与其余各解之间的最

小距离的标准差来衡量所得 Pareto 前端解集的分布情况.

$$S = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (\bar{d} - d_i)^2} \quad (7)$$

式中: d_i 为第 i 个个体距离目标空间中相邻点的最小曼哈顿距离, \bar{d} 为所得 d_i 的平均距离, n 为所得 Pareto 前端个体的数目. S 值越小,说明所求得解集的分布程度越均匀.

(3) 超体积 (Hyper Volume, HV)

超体积是用来计算所获 Pareto 解集个体在目标域所覆盖的体积^[16].利用该指标进行测试,可以实现对算法收敛性和多样性的综合评价.

$$HV = \text{volume}(\bigcup_{i=1}^Q v_i) \quad (8)$$

式中, Q 代表所获 Pareto 前端的数量, v_i 表示由参考原点和解 i 所形成的超体积,其值越大,表明所求解集越能广泛地覆盖在其真实前端上.

3.2 算法性能的对比与分析

为了进一步说明 MS-ACellDE 算法的性能表现,本文将其与 NSGA-II、SPEA2、MOEA/D^[17]以及 CellDE 进行了性能结果的参照对比.同时,为了减少随机因素的干扰,本文对各算法都进行了 30 次独立的运算,并对算法所需参数进行了合理统一的设定: NSGA-II、SPEA2 算法采用模拟二进制交叉、多项式变异实现进化,其中交叉概率为 0.9,变异概率设置为 $1/v$ (v 为相应决策变量的个数), $\eta_c = 20$, $\eta_m = 20$; MS-ACellDE、CellDE 算法采用差分进化策略,参数设置为: $CR = 0.1$, $F = 0.5$ ^[4]; MOEA/D 算法依据相关文献,参数设置为: $CR = 1.0$, $F = 0.5$, $p_m = 1/v$, $T = 10$ ^[18].为了更加直观的进行对比,上述算法将进化种群规模和外部种群规模的大小统一设置为 100,反馈个体的数目为 20.

为了使数据得到更为直观的展现,本文对算法所得最优值与次优值进行了不同形式的标注.表中灰色背景标注的部分表示为最优值,而字体加粗标注的部分表示为次优值.

表 1 为收敛性指标的计算结果.根据表格中的结果所示, MS-ACellDE 算法取得了 7 个函数中的 2 个最优值和 4 个次优值; MOEA/D 算法取得了 7 个函数中的 2 个最优值与 1 个次优值; CellDE 和 NSGA-II 算法均取得了 1 个最优值与 1 个次优值;而 SPEA2 算法只取得了一次最优值.在上述算法性能的比较中,本文所设计的 MS-ACellDE 算法展现出了比较优异的收敛性.尤其相比较于原始 CellDE 算法,改进后的算法在所设置的 7 个函数中 6 个函数都优于原始算法.该结果说明在算法设计过程中,相应种群数量的动态调整以及依据进化阶段而自适应变化的差分策略可以有效提高算法的收敛效率,使算法可以较好的逼近 Pareto 前端.

表 1 收敛性指标测试结果

测试函数		NSGA-II	SPEA2	MOEA/D	CellDE	MS-ACellDE
DTLZ1	平均值	1.092e-3	2.710e-3	1.095e-3	8.139e-4	6.386e-4
	标准差	6.966e-3	2.340e-3	6.313e-5	1.582e-4	2.806e-5
DTLZ2	平均值	1.123e-3	9.697e-4	5.758e-4	5.829e-4	4.543e-4
	标准差	7.742e-5	2.343e-4	1.522e-5	1.053e-4	1.066e-4
DTLZ3	平均值	2.400e-3	3.028e-3	1.197e-3	3.913e-1	2.870e-3
	标准差	1.229e-3	9.148e-4	3.603e-4	1.272e-2	1.367e-3
DTLZ4	平均值	1.148e-3	9.370e-4	7.302e-4	5.261e-4	4.916e-4
	标准差	1.802e-4	3.319e-4	1.067e-4	6.346e-5	1.722e-4
DTLZ5	平均值	4.930e-4	4.793e-4	3.974e-4	4.431e-4	4.414e-4
	标准差	3.647e-5	2.436e-5	3.179e-6	1.615e-5	2.348e-5
DTLZ6	平均值	7.020e-2	1.177e-2	4.580e-4	4.342e-4	4.401e-4
	标准差	3.904e-3	1.912e-3	1.914e-6	2.288e-5	2.214e-5
DTLZ7	平均值	2.036e-3	1.847e-3	5.416e-3	4.014e-3	2.804e-3
	标准差	3.423e-4	5.182e-4	2.308e-4	4.733e-4	3.050e-4

表 2 是算法所计算的分布性指标结果. 在所示的结果中, SPEA2 算法取得了 3 个最优值; CellDE 算法取得了 2 个最优值; MS-ACellDE 与 MOEA/D 算法各取得了一个最优值. 根据上述结果显示, 与 SPEA2 算法相比较而言, MS-ACellDE 算法在分布性能上仍然存在着不足, 需要得到进一步的加强与提高. 但是在整体表现上 MS-ACellDE 算法仍优于原始算法, 实现了分布性能的

提高, 而究其原因可以归纳为: (1) 在种群的进化过程中, MS-ACellDE 算法引入了更加丰富的邻居结构形式. 当多种邻居结构在算法中得到应用时, 可以有效的避免种群出现“早熟”的现象. (2) 在差分策略中, 添加了周期性的扰动策略, 使算法在整个进化过程都可以产生足够的扰动来促进其实现不断优化.

表 2 分布性指标测试结果

测试函数		NSGA-II	SPEA2	MOEA/D	CellDE	MS-ACellDE
DTLZ1	平均值	2.650e-2	2.307e-2	2.961e-2	2.100e-2	2.068e-2
	标准差	5.533e-3	5.426e-3	3.840e-4	1.329e-3	1.815e-3
DTLZ2	平均值	5.683e-2	2.390e-2	8.949e-2	5.186e-2	5.003e-2
	标准差	1.939e-3	2.153e-3	1.107e-3	2.282e-4	2.281e-3
DTLZ3	平均值	6.281e-2	7.435e-2	9.146e-2	2.924e-1	5.343e-2
	标准差	4.620e-3	3.99e-3	1.305e-3	5.613e-2	6.826e-3
DTLZ4	平均值	5.547e-2	2.299e-2	8.654e-2	5.051e-2	5.029e-2
	标准差	2.844e-3	1.859e-3	4.234e-3	2.018e-3	1.953e-3
DTLZ5	平均值	9.759e-3	4.537e-3	7.989e-3	9.271e-3	9.198e-3
	标准差	3.578e-4	2.600e-4	5.625e-5	5.732e-4	5.516e-4
DTLZ6	平均值	6.730e-2	1.737e-2	8.133e-3	9.988e-3	1.006e-2
	标准差	3.297e-3	1.770e-3	1.751e-5	4.626e-4	4.263e-4
DTLZ7	平均值	5.447e-2	2.994e-2	8.803e-2	6.894e-2	7.068e-2
	标准差	1.354e-2	3.948e-3	6.345e-3	7.679e-3	5.126e-3

表 3 展示了 5 种算法计算超体积所得的数值结果. 在表格中, 论文所设计的 MS-ACellDE 算法取得了 7 个函数中的 3 个最优值和 4 个次优值; SPEA2 算法取得了 7 个函数中的 3 个最优值; 原始的元胞差分算法取得了 7 个函数中的 1 个最优值和 3 个次优值. 从该结果中可

以看出, 本文所设计的基于多邻居结构的自适应元胞差分算法相比较于设计对照的 4 种算法而言, 其拥有更加优异的综合性能.

图 3 为五种算法在求解 DTLZ6 时所得的 Pareto 前端图.

表 3 综合性能指标测试结果

测试函数		NSGA-II	SPEA2	MOEA/D	CellDE	MS-ACellDE
DTLZ1	平均值	8.068e-1	8.136e-1	7.945e-1	8.152e-1	8.205e-1
	标准差	3.859e-3	7.633e-3	9.923e-4	3.838e-3	2.222e-3
DTLZ2	平均值	5.318e-1	5.529e-1	5.329e-1	5.376e-1	5.408e-1
	标准差	2.216e-3	1.786e-3	7.542e-4	3.247e-3	3.136e-3
DTLZ3	平均值	5.108e-1	5.587e-1	5.241e-1	0	5.397e-1
	标准差	1.345e-2	9.934e-4	4.385e-3	0	3.771e-3
DTLZ4	平均值	5.264e-1	5.327e-1	5.322e-1	5.374e-1	5.405e-1
	标准差	2.142e-2	9.692e-3	1.435e-3	3.238e-3	3.772e-3
DTLZ5	平均值	1.991e-1	1.987e-1	1.956e-1	1.996e-1	1.997e-1
	标准差	1.810e-4	1.011e-4	3.871e-5	7.837e-5	1.081e-4
DTLZ6	平均值	0	1.165e-1	1.959e-1	1.998e-1	1.997e-1
	标准差	0	2.586e-2	9.762e-2	9.359e-5	1.078e-4
DTLZ7	平均值	2.637e-1	2.716e-1	2.193e-1	2.617e-1	2.711e-1
	标准差	1.303e-2	1.381e-2	4.603e-3	2.244e-3	7.898e-4

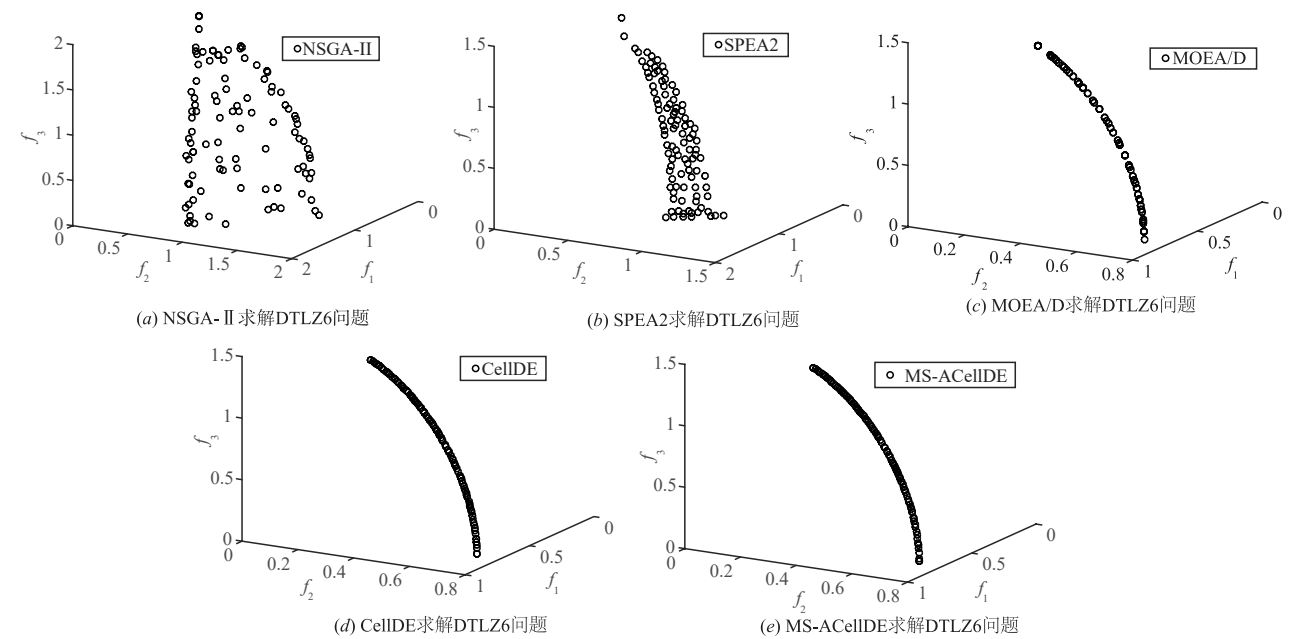


图 3 五种算法在求解 DTLZ6 时所获 Pareto 前端图

从图中可以得出,MS-ACellDE 与 CellDE 算法相较于其余三种算法,两者表现出更加优异的收敛性能.而在 CellDE 与 MS-ACellDE 之间,两者在收敛性能与分布性能上没有明显的差异.但是结合两种算法在求解其余测试函数时所获得的数据,本文所设计的 MS-ACellDE 算法展现出了更加优异的性能表现,说明其在求解 Pareto 前端时具有明显的性能优势.

4 结束语

本文在传统元胞差分算法的基础上,针对其进化过程的复杂需求,对算法进行了自适应改进.为了避免出现种群个体早熟的现象,算法根据个体的性能优劣对其进行了不同邻居结构的分配.一方面,通过邻居结构的调整,加强了种群个体之间的交流.另一方面,保留了传统邻居结构,保证算法的局部开发性能.并且在进化过程中,算法通过不同变异策略的调整实现了全局搜索能力与局部寻优能力的进一步平衡.

参考文献

- [1] 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.
- [2] BLEULER S, BRACK M, THIELE L, et al. Multiobjective genetic programming: Reducing bloat using SPEA2 [A]. Proceedings of the 2001 Congress on Evolutionary Computation [C]. Seoul: IEEE, 2001. 536–543.
- [3] COELLO C A C, PULIDO G T, LECHUGA M S. Handling multiple objectives with particle swarm optimization [J]. IEEE Transactions on Evolutionary Computation, 2004, 8(3): 256–279.
- [4] DURILLO J J, NEBRO A J, LUNA F, et al. Solving three-objective optimization problems using a new hybrid cellular genetic algorithm [A]. Parallel Problem Solving from Nature-PPSN X [C]. Berlin Heidelberg: Springer, 2008. 661–670.
- [5] 张屹, 郑小东, 万兴余, 等. 基于差分元胞多目标遗传算法的动压滑动轴承优化设计 [J]. 机械传动, 2014, 038(9): 64–68.
ZHANG Yi, ZHENG Xiao-dong, WAN Xing-yu, et al. Optimization design of hydrodynamic sliding bearing based on differential cellular genetic algorithm [J]. Journal of Mechanical Transmission, 2014, 038(9): 64–68.
- [6] 王亚良, 倪晨迪, 曹海涛, 等. 两阶段动态差分智能元胞机算法 [J]. 计算机集成制造系统, 2020, 26(4): 989–1000.
- WANG Ya-liang, NI Chen-di, CAO Hai-tao, et al. Two-stage dynamic differential agent cellular automata algorithm [J]. Computer Integrated Manufacturing Systems, 2020, 26(4): 989–1000. (in Chinese)
- [7] 詹腾, 张屹, 朱大林, 等. 基于多策略差分进化的元胞多目标遗传算法 [J]. 计算机集成制造系统, 2014, 20(6): 1342–1341.
ZHAN Teng, ZHANG Yi, ZHU Da-lin, et al. Cellular multi-objective genetic algorithm based on multi-strategy differential evolution [J]. Computer Integrated Manufacturing Systems, 2014, 20(6): 1342–1351. (in Chinese)
- [8] 王福才, 周鲁萍. 混合精英策略的元胞多目标遗传算法及其应用 [J]. 电子学报, 2016, 44(3): 709–719.
WANG Fu-cai, ZHOU Lu-ping. Cellular multi-objective genetic algorithm based on hybrid elite and application [J]. Acta Electronica Sinica, 2016, 44(3): 709–719. (in Chinese)
- [9] DORRONSORO B, BOUVRY P. Cellular genetic algorithms without additional parameters [J]. The Journal of Supercomputing, 2013, 63(3): 816–835.
- [10] ALI M Z, AWAD N, SUGANTHAN P N, et al. An adaptive multipopulation differential evolution with dynamic population reduction [J]. IEEE Transactions on Cybernetics, 2017, 47(9): 2768–2779.
- [11] ALBA E, DORRONSORO B. The exploration/exploitation tradeoff in dynamic cellular genetic algorithms [J]. IEEE Transactions on Evolutionary Computation, 2005, 9(2): 126–142.
- [12] TANG L, DONG Y, LIU J. Differential evolution with an individual-dependent mechanism [J]. IEEE Transactions on Evolutionary Computation, 2015, 19(4): 560–574.
- [13] WU G, MALLIPEDDI R, SUGANTHAN P N, et al. Differential evolution with multi-population based ensemble of mutation strategies [J]. Information Sciences, 2016, 329: 329–345.
- [14] XU B, TAO L, CHEN X, et al. Adaptive differential evolution with multi-population-based mutation operators for constrained optimization [J]. Soft Computing, 2019, 23: 3423–3447.
- [15] DEB K, THIELE L, LAUMANN S M, et al. Scalable multi-objective optimization test problems [A]. Proceedings of the 2002 Congress on Evolutionary Computation [C]. Honolulu, USA: IEEE, 2002. 825–830.
- [16] ZITZLER E, THIELE L. Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach [J]. IEEE Transactions on Evolutionary Computation, 1999, 3(4): 257–271.

- [17] ZHANG Q, LI H. MOEA/D: A multiobjective evolutionary algorithm based on decomposition [J]. IEEE Transactions on Evolutionary Computation, 2007, 11(6): 712–731.
- [18] LI K, DEB K, ZAHNG Q, et al. An evolutionary many-objective optimization algorithm based on dominance and decomposition [J]. IEEE Transactions on Evolutionary Computation, 2015, 19(5): 694–716.

作者简介



王亚良 男, 1977 年生于浙江绍兴. 现为浙江工业大学机械工程学院高级实验师. 主要研究方向为生产系统优化和智能制造.
E-mail: wangyaliang@zjut.edu.cn



倪晨迪 男, 1995 年生于浙江绍兴. 现为浙江工业大学机械工程学院硕士研究生. 主要研究方向为智能制造和优化算法.
E-mail: 2060662618@qq.com



金寿松(通信作者) 男, 1965 年生于浙江东阳. 现为浙江工业大学机械工程学院副教授. 主要研究方向为机械设计和智能制造.
E-mail: jinshs@zjut.edu.cn