

基于自适应多种群策略的混合多目标优化算法

付亚平, 王洪峰, 黄敏, 王兴伟

(东北大学信息科学与工程学院; 流程工业综合自动化国家重点实验室, 沈阳 辽宁 110819)

摘要: 为了能够快速准确地获得多目标优化问题的一组具有较好质量和分布性的非支配解, 提出了一种自适应多种群混合多目标优化算法. 该算法将多目标优化问题分解为多个单目标子问题, 在每次迭代时, 根据种群在目标空间和解空间的分布情况为多个子问题分别构造子种群, 并采用粒子群优化算法对子问题最优解实施搜索, 利用差分进化算法对外部档案实施进化. 通过对标准测试函数仿真实验, 并与经典的及类似策略的多目标优化算法进行比较, 结果表明所提出的算法能够利用较少的估值次数获得较好质量和分布性的非支配解集.

关键词: 多种群; 多目标优化算法; 自适应; 分解方法

中图分类号: TP273

文献标识码: A

文章编号: 1000-5781(2017)06-0738-11

doi: 10.13383/j.cnki.jse.2017.06.003

Adaptive multipopulation strategy based hybrid multiobjective optimization algorithm

Fu Yaping, Wang Hongfeng, Huang Min, Wang Xingwei

(College of Information Science and Engineering, Northeastern University; State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang 110819, China)

Abstract: In order to obtain a nondominated solution set with higher quality and better distribution quickly and accurately, an adaptive multipopulation strategy based hybrid multiobjective optimization algorithm is proposed. The proposed algorithm decomposes the multiobjective optimization problem into multiple single objective subproblems, and multiple subpopulations are constructed for each subproblem according to the distribution of population in the objective and solution space at each iteration. The particle swarm optimization is used to search the optimal solutions of subproblems, and the external archive is evolved by the differential evolution algorithm. The simulation of several benchmark test functions shows that, compared with the state-of-art and similar multiobjective optimization algorithms, the proposed algorithm can obtain better nondominated solutions with less fitness evaluation times.

Key words: multipopulation; multiobjective optimization algorithm; adaptive; decomposition approach

1 引言

科学研究与工程应用的许多优化问题都存在两个或多个相互冲突的目标, 这类问题被称为多目标优化问题. 多目标优化问题通常不存在一个能使所有目标同时达到最优的解, 于是多目标优化算法的目的是获得

收稿日期: 2014-12-30; 修订日期: 2015-12-03.

基金项目: 国家杰出青年科学基金资助项目(71325002); 国家自然科学基金重点国际合作研究资助项目(71620107003); 国家自然科学基金创新研究群体资助项目(61621004); 国家自然科学基金资助项目(71671032); 山东省自然科学基金项目(ZR2016FP02).

一个质量较好且分布均匀的非支配解集^[1]. 基于群体智能理论的优化算法可以在一次运行获得多个解, 特别适合求解多目标优化问题. 近年来, 众多学者对这类多目标优化算法进行了广泛而深入的研究, 并提出了许多经典的优化算法, 包括基于支配规则的多目标优化算法, 基于指标的多目标优化算法以及基于分解的多目标优化算法. 其中, 基于支配规则的多目标优化算法采用个体间的支配关系及适应度共享机制确定个体的优劣^[2-5]. 基于指标的多目标优化算法考虑个体与最优解集的逼近程度及分布性, 通过计算个体对某一指标值的贡献度评价个体的优劣^[6,7]. 基于分解的多目标优化算法首先将多目标优化问题分解为多个单目标子问题, 然后采用优化算法同时求解这些子问题^[8-12].

近年来, 采用多种群策略求解多目标优化问题的研究思路引起了学者们的重视. Jaimes 等^[13]指出多种群策略不仅可以提高种群的多样性, 同时可以容易地与其他搜索技术结合以提高算法的搜索效率. 采用多种群策略的多目标优化算法可以分为基于目标向量的多种群多目标优化算法, 基于决策变量的多种群多目标优化算法, 基于分解的多种群多目标优化算法以及基于种群多样性的多种群多目标优化算法. 其中, 基于目标向量的多种群多目标优化算法为每一个目标分量分别构造子种群, 通过子种群的进化求解多目标优化问题^[14]; 基于决策变量的多种群多目标优化算法为每一个决策变量构造子种群, 通过子种群之间的协同进化完成对解空间的搜索^[15]; 基于分解的多种群多目标优化算法将多目标优化问题分解为多个单目标子问题, 通过为这些子问题构造子种群的方式求解多目标优化问题^[16]. 这三种多种群策略为子种群分配了确定的搜索方向, 并在搜索过程中保持子种群的组成成员不变, 这有助于改善算法的收敛性. 基于种群多样性的多种群多目标优化算法在搜索过程中周期性的改变子种群成员个体, 这有助于维持种群的多样性^[17]. 考虑到适当地平衡和控制多种群的收敛性和多样性有助于求解多目标优化问题, 这就使得设计一种自适应的多种群策略求解多目标优化问题成为一个值得深入研究的工作.

上述算法在求解多目标优化问题时使种群整体逐渐逼近于最优前沿, 以获得一个近似最优的非支配解集. 而基于群体的优化算法在搜索过程中, 优势个体通常会吸引其他个体快速向其所在区域移动, 这有利于加快算法收敛速度. 为此, 本文考虑首先获得多个均匀分布的优势解, 然后在这些优势解的引导下快速地搜索其他非支配最优解. 基于这个研究思路, 本文提出了一种基于自适应多种群的混合多目标优化算法(adaptive multipopulation based hybrid multiobjective optimization algorithm, AMHMOA), 该算法将多目标优化问题分解为多个单目标子问题, 并分阶段的引入这些子问题进行求解. 在每次迭代时, 根据种群在目标空间和解空间的分布情况, 选择各子问题的当前最好个体和与其相似的个体分别为这些子问题构造子种群, 并采用粒子群优化算法对这些子问题进行并行求解; 采用差分进化算法对外部档案的非支配解实施进化以进一步提炼和扩充非支配解集. 为保持种群的多样性以获得一个分布均匀的非支配解集, 粒子在搜索过程中采用变异策略和邻域学习策略; 而差分进化的变异操作采用全局或局部策略, 种群与外部档案间实施信息交互机制以提高种群质量及加快算法收敛速度. 图 1 给出了该算法种群和外部档案的优化工具及搜索策略的示意图. 通过对十二个典型的测试函数进行仿真实验, 并与几种典型的多目标优化算法进行对比, 结果表明所提出的算法能够在较少的估值次数条件下, 获得的非支配解集在解的数量, 质量及分布性方面具有较好的效果.

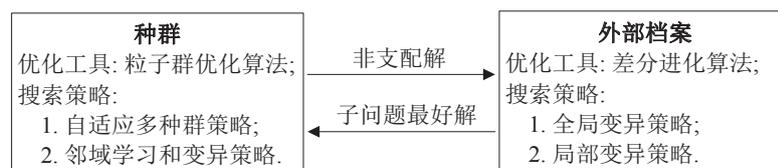


图 1 种群与外部档案的优化工具及搜索策略

Fig. 1 Optimization tool and search strategy for population and external archive evolution

2 AMHMOA 算法的设计

本文所提出的自适应多种群混合多目标优化算法的执行过程描述如下. 在完成算法参数初始化和问题分解(如2.1节)后, 算法将进入下述迭代过程: 首先引入多个均匀分布的子问题(如2.2节); 然后从种群中选择各子问题的最好解并为各子问题分别构造子种群(如第2.3节), 采用外部档案对当前各子问题最好解实施更新(如第2.4节); 其次对种群和外部档案分别采用粒子群优化和差分进化算法实施搜索(如第2.4和2.5节), 并对外部档案实施更新(如第2.5节); 最后将这些子种群合并, 并进入下一次迭代过程. 该算法结合多目标优化的分解方法, 提出了一种自适应多种群策略, 并设计了多种群的搜索策略和外部档案的进化策略. 需要解决的关键问题有: 多目标优化问题的分解, 子问题的分阶段引入方法, 多种群的构造方法, 多种群的搜索策略及外部档案的进化策略, 下面将详细描述这些关键问题的解决方法.

2.1 多目标优化问题的分解

本文所提出的算法将多目标优化问题分解为多个单目标问题进行求解, 其中切比雪夫方法和加权求和方法是常用的分解方法^[18]. 切比雪夫方法不依赖于问题特点, 能够求解前沿面为凸或非凸形状的多目标优化问题. 因此, 本文采用切比雪夫方法将多目标优化问题分解为多个单目标子问题. 令 $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_m)^T$ 为权重向量, $\sum_{i=1}^m \lambda_i = 1, \lambda_i \geq 0, z^* = (z_1^*, z_2^*, \dots, z_m^*)^T$ 表示理想点, 对于最小化问题, $z_i^* = \min\{f_i(x) \mid x \in \Omega\}, i = 1, 2, \dots, m$, 其中 m 为目标数, Ω 为解空间, 则子问题为

$$\begin{cases} \text{Min } g(x \mid \lambda, z^*) = \max_{1 \leq i \leq m} \{\lambda_i | f_i(x) - z_i^* | \} \\ \text{s.t. } x \in \Omega. \end{cases} \quad (1)$$

对于给定的 λ , 式(1)存在唯一的最优解, 也是原多目标优化问题的非支配最优解. 因此, 可以通过设定多个权重向量构建多个子问题, 并采用多种群策略对多个子问题进行求解. 这里权重向量与子问题为一一对应关系.

2.2 子问题的分阶段引入方法

采用上述方法产生一个子问题集合, 并在算法执行过程中分阶段地引入这些子问题进行求解. 算法在执行时, 首先将所有子问题均匀分组, 图2给出了将一个两目标问题分解为十个子问题并分成三组的实例,

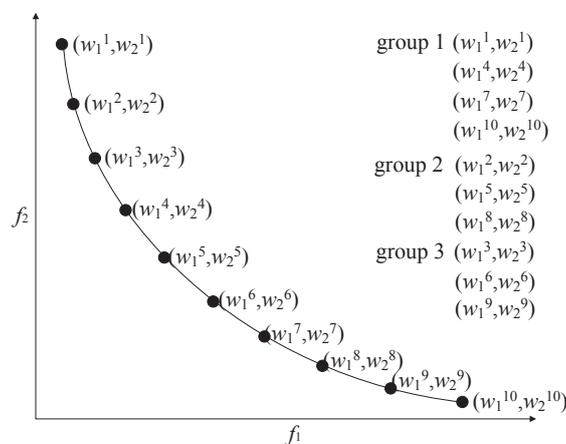


图2 子问题的分组方法

Fig. 2 The grouping method for subproblems

第一组包含子问题(权重向量)1, 4, 7, 10, 第二组包含子问题 2, 5, 8, 第三组包含子问题 3, 6, 9. 算法初始

阶段引入第一组子问题进行求解; 在经过设定的估值次数后将第二组子问题引入, 此时第一组和第二组同为当前求解的子问题, 为保持种群质量, 这时将种群与外部档案合并, 若合并后种群规模小于所需个体数量, 则随机生成其他个体, 否则, 采用非支配排序和拥挤距离评估方法剔除多余个体. 第三组子问题按照相同的方法引入. 这种子问题分阶段引入的方法, 可以有效地利用种群中优势解的信息, 为新子问题构造质量较高的子种群, 从而提高对其最优解的搜索效率. 子问题的分阶段引入方法如下.

步骤 1 将一组均匀分布的子问题加入到 W_{st} , 其中 W_{st} 表示当前求解的子问题集合, 令 n_1 表示当前求解的子问题数, n_2 表示每个子种群包含的个体数量;

步骤 2 将当前种群 P 与外部档案中的个体进行合并, 并对当前种群进行调整;

步骤 2.1 若 $|P| < n_1 \times n_2$, 则随机生成 $(n_1 \times n_2 - |P|)$ 个新个体;

步骤 2.2 若 $|P| > n_1 \times n_2$, 则采用分层和拥挤距离方法从 P 中删除 $(|P| - n_1 \times n_2)$ 个劣势个体;

步骤 3 输出 W_{st} 并退出.

2.3 多种群的构造方法

本文设计了一种自适应多种群策略, 算法在每次迭代时, 根据种群在目标空间和解空间的分布情况自适应地为当前求解的子问题分别构造子种群. 该方法在目标空间内选择各子问题的最好解, 并在解空间内选择与最好解相似的个体为各子问题分别构造子种群进行求解. 这种自适应构造多种群的策略有利于提高对子问题最优解的求解精度, 能够快速获得子问题的近似最优解. 这里规定子种群的个体互不重合, 即一个个体仅属于一个子种群. 子问题最好解的选择及子种群的构造方法如下.

通过分解方法获得的多个子问题通常被认为同等重要, 而极端子问题 $(\sum_{i=1}^m \lambda_i = 1, \text{且 } \exists i \in [1, m] \text{ 满足 } \lambda_i = 1)$ 会优先占有极端目标上具有优势的个体, 可能占用其他子问题的当前最好个体. 为解决该问题采用如下两种策略: 1) 随机选择子问题并为其选择最好个体; 2) 极端点的权重向量中分量为 0 的权重设为较小数, 如 10^{-4} , 以降低极端目标的选择压力. 子问题最好解选择方法如下.

步骤 1 将 W_{st} 中的子问题进行随机排序, $i \leftarrow 1, S \leftarrow \emptyset$, 其中 S 为当前求解的子问题的最好解集合;

步骤 2 选择 W_{st} 中的第 i 个子问题(权重向量);

步骤 3 对于种群 P 中的所有个体, 计算其对于第 i 个子问题的目标值, 并选择该子问题的最优个体 p , $P \leftarrow P - \{p\}, S \leftarrow S + \{p\}, i \leftarrow i + 1$;

步骤 4 若 $i \leq n_1$, 则转步骤 2; 否则输出 S 并退出.

在确定各子问题的最好解之后, 需要评价种群中未被处理的个体与子问题最好解的相似性以为该子问题构造子种群. 本文所提出的算法面向连续多目标优化问题, 因此, 采用欧式距离评价不同个体的相似性. 这种评价方法已被应用于求解多峰优化问题^[19], 如式(2)所示. 其中 p 和 q 表示两个不同的个体, n 为决策变量的维数, $\{x_{p1}, x_{p2}, \dots, x_{pn}\}$ 和 $\{x_{q1}, x_{q2}, \dots, x_{qn}\}$ 表示解空间内个体 p 和 q 的位置. $\text{Sim}(p, q)$ 为个体 p 和 q 的相似程度, 其值越小, 表示两个个体的相似性越高. 依据欧式距离的评价方法对种群中未被处理的个体与子问题最好解进行相似性评估, 并将这些相似性高的个体及子问题最好解划分为子种群. 按照该方法依次为各子问题构造子种群. 这种在解空间内将与子问题最好解距离较近的个体划分为子种群的方法可以提高对子问题最优解的搜索精度, 也加快了寻优速度. 多种群构造方法如下.

步骤 1 $i \leftarrow 1, s_i \leftarrow \emptyset$, 其中 s_i 为第 i 个子问题的子种群包含的子问题集合;

步骤 2 从 S 中选择第 i 个个体 $s, s_i \leftarrow s_i + \{s\}$;

步骤 3 对于 P 中的所有个体, 计算其与 s 的相似性;

步骤 4 将与 s 相似性较高的 $n_2 - 1$ 个个体置于 s_i 中, 并从 P 中剔除这些个体;

步骤 5 $i \leftarrow i + 1$, 若 $i \leq n_1$, 则转步骤 2; 否则输出各子问题对应的子种群并退出.

$$\text{Sim}(p, q) = \sqrt{\sum_{i=1}^n (x_{pi} - x_{qi})^2} \quad (2)$$

图3以三个子问题为例,给出了构造多种群的示意图.首先在目标空间内按照一定顺序为子问题1,子问题2及子问题3分别选择最好解 s_1, s_2 及 s_3 ;然后在解空间内选择与各最好解相似性较高的个体为各子问题构造子种群.

2.4 多种群的搜索策略

粒子群优化算法是一种模仿鸟群觅食的群体智能优化算法,具有很快的收敛性能,这也使得种群易失去多样性.学者们提出了诸多的改进措施以提高该算法求解多目标优化问题的性能,如变异策略^[20],参数自适应调整策略^[21]及多样化学习策略^[22].

本文采用粒子群算法对各子问题最优解进行搜索,子种群的个体随机选择子种群最好个体或近邻子种群最好个体进行学习,并采用文献[23]提出的变异方法对个体实施变异操作.这里的近邻关系定义为目标空间内权重向量的欧式距离,距离越小表示子种群距离越近.为提高算法的求解性能,种群与外部档案之间实施信息交互机制,如图4所示.这种交互主要体现在以下方面:a.将种群搜索到的非支配解更新外部档案;b.利用外部档案中的非支配解更新子问题的最好解;c.当有新的子问题引入时;d.从外部档案中引入新个体以构造更多的子种群.

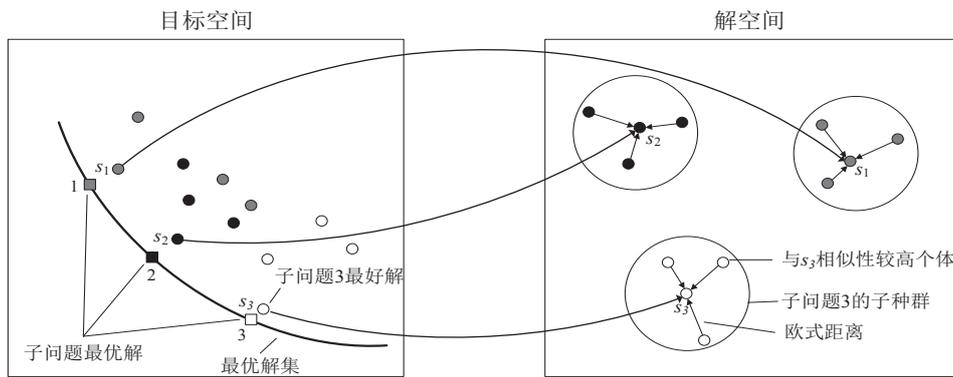


图3 多种群构造示意图

Fig. 3 Illustration of constructing multipopulation

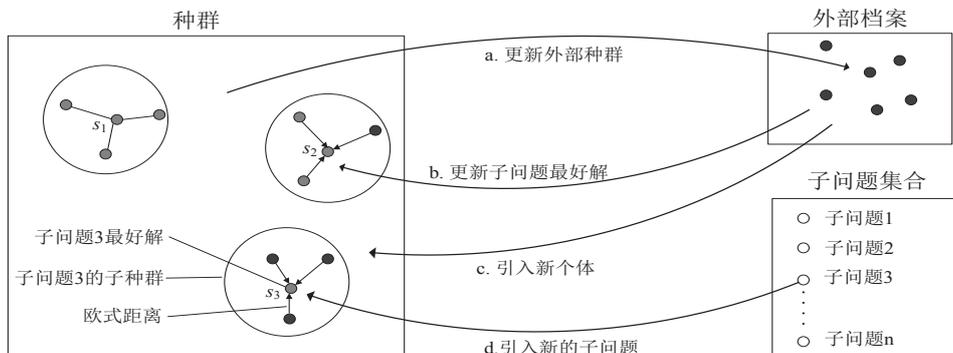


图4 种群与外部档案的信息交互机制

Fig. 4 Information interaction mechanism between population and external archive

2.5 外部档案的进化策略

外部档案用以保存搜索过程中发现的非支配解, 这里采用支配规则对外部档案实施更新^[3]. 外部档案的最大容量为 100, 若外部档案的规模超过其容量, 则采用拥挤距离方法剔除多余的劣势解. 为快速扩充和改进外部档案中的非支配解集, 采用差分进化算法对外部档案实施进化. 当外部种群规模较小时采用全局策略, 对所有个体均执行差分进化操作, 变异个体在整个外部档案中随机选择, 这种策略可以快速扩充外部档案. 而当外部种群规模较大时采用局部策略, 首先计算整个外部档案所有个体的拥挤距离, 然后采用轮盘赌方法选择较稀疏区域的个体执行差分进化操作, 并在该个体的局部区域内选择变异个体执行变异操作, 这种策略可以提高外部档案中非支配解集的质量和分布性.

3 仿真实验与分析

为验证本文所提出的算法在求解连续多目标优化问题的性能, 从文献[24]中选择十二个测试函数进行实验, 其中有七个两目标测试函数, 五个三目标测试函数. 实验选择 NSGAI^[2], MOEA/D^[8], SMSEMOA^[7], TVMOPSO^[21] 及 DMSPSO^[17] 做为比较算法. 其中, NSGAI 是基于支配规则的多目标优化算法; MOEA/D 是基于分解的多目标优化算法; SMSEMOA 是基于指标的多目标优化算法; TVMOPSO 是采用参数自适应策略的多目标粒子群优化算法; DMSPSO 同样是采用构造子种群的多目标粒子群优化算法.

3.1 参数设置

本文所提出算法采用文献[8]的方法生成含有 100 个权重向量(子问题)的集合, 初始种群规模为 50, 初始子问题数量为 10, 子种群规模为 5; 算法每经过 1 000 次估值后增加九个子问题, 并将子种群规模调整为 3, 个体等概率的选择自身种群最好或近邻范围内的子种群最好个体学习, 近邻范围为 $0.2 \times n_1$. 比较算法的参数设置均同原文献. 为验证本文所提出的算法与同类算法相比具有更快的收敛性能, 以 5 000 次估值作为停止条件. 实验采用所提出的算法及比较算法对每一测试函数分别求解 30 次, 并取均值和方差作为评价标准. 为更好的比较实验结果, 采用自由度为 58, 显著性水平为 0.05 的 t 检验方法, 符号“+”, “-”, “~”分别表示本文所提出的算法性能显著优于, 显著差于及基本等同于其比较算法.

为验证本文所提出算法所获得非支配解集的收敛性和分布性, 选择数量指标^[25], IGD 指标^[26]及 Hypervolume 指标^[26]评价算法的性能, 分别说明如下.

1) 数量指标: 该指标用于统计算法获得的非支配解的数量.

2) IGD 指标: 该指标可以综合评价所获得的非支配解集的收敛性和分布性,

$$\text{IGD}(\bar{P}, P^*) = \sum_{v \in P^*} d(v, \bar{P}) / |P^*|, \quad (3)$$

其中 P^* 为真实最优非支配解集的一组均匀采样, \bar{P} 为由多目标优化算法获得的一组非支配解集, $d(v, \bar{P})$ 为非支配最优解 v 与非支配解集 \bar{P} 中最近点的欧式距离; $|P^*|$ 表示解集 P^* 中非支配最优解的个数.

3) Hypervolume 指标: 该指标计算非支配解集中的所有非支配解与参考点围成的超立方体的体积, 其可以综合评价非支配解集的收敛性和分布性,

$$\text{Hypervolume} = \text{volume} \left(\bigcup_{i=1}^{|\bar{P}|} \text{HyperRectangle}(p^i, r) \right), \quad (4)$$

其中 r 表示参考点, 且满足 \bar{P} 中的所有非支配解均支配 r , $\text{HyperRectangle}(p^i, r)$ 表示 p^i 与 r 围成的区域, $\text{HyperRectangle}(p^i, r) = \{v \mid p_j^i \leq v_j \leq r_j\}$, $j = 1, 2, \dots, m$, 该指标值越大, 表示算法性能越好. 由于该指标与参考点的选择有关, 本文将所有算法获得的指标值进行规范化处理后进行比较.

3.2 实验结果与分析

首先验证本文所提出的算法及比较算法在获得的非支配解数量方面的性能, 表 1 给出了算法的数量指

标的实验结果.

表 1 六种算法的数量指标的实验结果比较

Table 1 Comparison of six algorithms via quantity-metric

Instance	AMHMOA	NSGAI	MOEA/D	SMSEMOA	TVMOPSO	DMSPSO
FON	100±0.0	98±1.6(+)	79±13.8(+)	100±0.0(∼)	100±0.0(∼)	100±0.0(∼)
KUR	100±0.0	98±2.3(+)	84±7.1(+)	100±0.0(∼)	23±19.4(+)	41±133.8(+)
ZDT1	100±0.0	58±96.2(+)	28±69.1(+)	63±108.5(+)	100±0.0(∼)	15±13.9(+)
ZDT2	98±6.5	16±95.4(+)	9±60.8(+)	13±199.5(+)	100±0.0(∼)	9±16.0(+)
ZDT3	100±0.0	74±99.5(+)	25±19.5(+)	80±104.4(+)	98±109.3(∼)	15±6.3(+)
ZDT4	100±0.0	23±132.2(+)	74±207.4(+)	82±680.2(+)	91±466.8(+)	8±15.7(+)
ZDT6	100±0.0	17±6.4(+)	47±91.5(+)	14±15.4(+)	100±0.0(∼)	42±850.9(+)
DTLZ2	100±0.0	100±0.0(∼)	98±5.4(∼)	100±0.0(∼)	100±0.0(∼)	100±0.0(∼)
DTLZ4	100±0.0	100±0.0(∼)	100±0.0(∼)	97±326.7(+)	23±17.8(+)	14±25.7(+)
DTLZ5	100±0.0	100±0.0(∼)	100±0.0(∼)	100±0.0(∼)	100±0.0(∼)	100±0.0(∼)
DTLZ6	100±0.0	100±0.0(∼)	100±0.0(∼)	100±0.0(∼)	100±0.0(∼)	100±0.0(∼)
DTLZ7	100±0.0	100±0.0(∼)	100±0.0(∼)	100±0.0(∼)	100±0.0(+)	67±926.9(∼)

从表中可以看出, 本文所提出的算法能够获得较多的非支配解, 其在十一个测试函数上能够获得指定数量的非支配解(外部档案的最大容量), 表明所提出的算法能够用较少的估值次数快速地获得所需数量的非支配解, 其他算法均在某些测试函数上无法获得指定数量的非支配解. 从方差上看, 所提出的算法的波动性较小, 具有较好的稳定性.

其次, 验证所提出的算法在 IGD 指标方面的性能, 表 2 给出了 IGD 指标的实验结果.

表 2 六种算法的 IGD 指标的实验结果比较

Table 2 Comparison of six algorithms via IGD-metric

Instance	AMHMOA	NSGAI	MOEA/D	SMSEMOA	TVMOPSO	DMSPSO
FON	0.004 7±0.000 1	0.005 3±0.000 0(∼)	0.005 0±0.000 0(∼)	0.005 3±0.000 0(∼)	0.016 3±0.000 0(+)	0.010 2±0.000 0(+)
KUR	0.023 9±0.001 6	0.042 8±0.000 0(+)	0.047 9±0.000 0(+)	0.041 4±0.000 0(+)	0.419 7±0.011 2(+)	0.250 1±0.006 8(+)
ZDT1	0.012 9±0.001 3	0.164 9±0.002 0(+)	0.386 5±0.974 4(+)	0.119 2±0.002 1(+)	0.014 5±0.000 0(∼)	0.169 9±0.001 9(+)
ZDT2	0.013 3±0.000 0	0.605 8±0.056 9(+)	0.446 8±0.031 1(+)	0.551 6±0.046 0(+)	0.050 0±0.014 5(∼)	0.220 9±0.017 4(+)
ZDT3	0.036 6±0.002 4	0.128 8±0.000 5(+)	0.186 9±0.004 2(+)	0.098 8±0.000 3(+)	0.057 3±0.000 1(+)	0.198 0±0.001 2(+)
ZDT4	0.049 0±0.000 3	0.878 0±0.114 1(+)	2.031 9±0.665 8(+)	1.213 8±0.335 0(+)	0.429 9±0.234 9(+)	1.406 6±0.869 7(+)
ZDT6	0.003 5±0.000 0	1.779 1±0.054 8(+)	0.060 8±0.000 2(+)	0.885 3±0.016 4(+)	0.003 6±0.000 0(+)	0.0575±0.001 9(+)
DTLZ2	0.654 7±0.000 0	0.670 8±0.000 0(+)	0.661 0±0.000 0(∼)	0.661 2±0.000 0(∼)	0.785 6±0.001 1(+)	0.793 6±0.004 9(+)
DTLZ4	0.081 2±0.008 7	0.185 5±0.045 5(+)	0.700 3±0.063 8(+)	0.317 3±0.057 4(+)	0.415 7±0.002 7(+)	0.453 6±0.007 8(+)
DTLZ5	0.282 4±0.000 5	0.342 9±0.002 3(+)	0.300 3±0.000 0(+)	0.300 4±0.000 2(+)	0.359 3±0.001 4(+)	0.438 3±0.002 2(+)
DTLZ6	0.311 1±0.000 6	4.640 7±0.029 0(+)	3.449 2±0.261 8(+)	3.751 2±0.042 0(+)	1.095 5±0.256 5(+)	3.527 7±0.708 4(+)
DTLZ7	0.075 1±0.000 0	0.124 7±0.005 4(+)	0.313 3±0.034 3(+)	0.116 4±0.011 5(+)	0.076 8±0.000 0(∼)	0.220 2±0.008 0(+)

从表中的实验结果可以发现, 本文所提出的算法在十二个测试函数的优化效果均优于 NSGAI, MOEA/D, SMSEMOA, TVMOPSO 及 DMSPSO; t 检验的结果表明, 本文所提出的算法在 KUR, ZDT1, ZDT3, ZDT4, ZDT6, DTLZ4, DTLZ5 及 DTLZ6 上的优化效果均显著优于其所有比较算法. 另外, 本文所提出的算法在十一个测试函数上的优化效果要显著优于 NSGAI, 在十个测试函数上的优化效果要显著优于 MOEA/D 和 SMSEMOA, 在九个测试函数上的优化效果要显著优于 TVMOPSO, 在所有测试函数上的优化效果均显著优于 DMSPSO. 从方差上看, 所提出的算法在各测试函数上的优化效果波动性较小, 稳定性较好. 图 5 及图 6 给出了所提出的算法及比较算法在 30 次求解 ZDT4 及 DTLZ6 实验结果的盒状图, 从图中可以发现, 本文所提出的算法 30 次运行的离散程度较小, 具有较好的稳定性. 通过上述分析, 可以发现本文所提出的算法在 IGD 指标方面能够取得较好的效果, 表明所提出的算法能够获得较好质量和分布性的非支配解集.

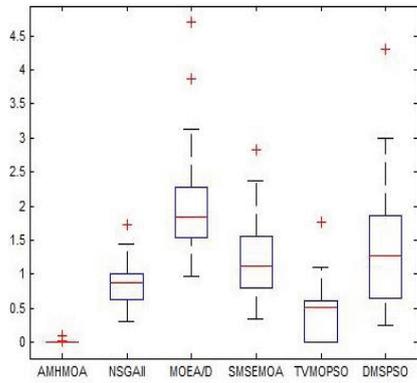


图 5 ZDT4 测试函数的 30 次运行 IGD 指标实验结果的盒状图
Fig. 5 Boxplot of experimental result via IGD-metric by 30 runs for ZDT4

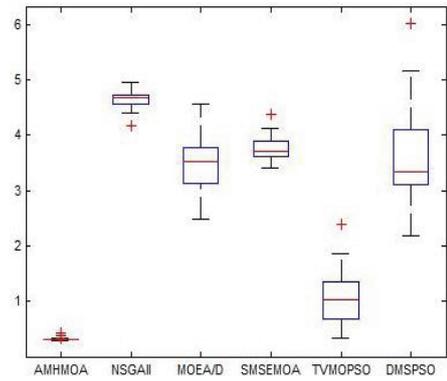


图 6 DTLZ6 测试函数的 30 次运行 IGD 指标实验结果的盒状图
Fig. 6 Boxplot of experimental result via IGD-metric by 30 runs for DTLZ6

最后, 验证本文所提出的算法在 Hypervolume 指标方面的性能, 表3给出了 Hypervolume 指标的实验结果. 从表中的实验结果可以发现, 本文所提出的算法在 ZDT2, ZDT4, DTLZ5 及 DTLZ6 上的优化效果均显著优于其所有比较算法. 与 NSGAI 和 MOEA/D 相比, 所提出的算法在十二个测试函数上的优化效果均要优于 NSGAI 和 MOEA/D, 且有十一个测试函数的优化效果显著优于 NSGAI, 有十个测试函数显著优于 MOEA/D; 与 SMSEMOA 和 TVMOPSO 相比, 在十一个测试函数的优化效果优于 SMSEMOA 和 TVMOPSO, 且有九个测试函数的优化效果显著优于 SMSEMOA, 有七个测试函数的优化效果显著优于 TVMOPSO; 与 DMSPSO 相比, 所提出的算法在十二个测试函数的优化效果均优于 DMSPSO, 且有十一个测试函数的优化效果显著优于 DMSPSO. 从方差上看, 所提出的算法在各测试函数的优化效果波动性较小, 稳定性较好. 图 7 及图 8 分别给出了所提出的算法及比较算法 30 次求解 ZDT2 及 DTLZ6 实验结果的盒状图, 从图中可以发现, 本文所提出的算法 30 次运行的离散程度较小, 具有较好的稳定性. 通过上述分析, 可以发现本文所提出的算法在 Hypervolume 方面具有较好的性能, 表明所提出的算法能够获得质量和分布性较好的非支配解集.

表 3 六种算法的 Hypervolume 指标比较
Table 3 Comparison of six algorithms via Hypervolume-metric

Instance	AMHMOA	NSGAI	MOEA/D	SMSEMOA	TVMOPSO	DMSPSO
FON	0.355 5±0.000 2	0.335 3±0.000 1(+)	0.168 6±0.000 0(+)	0.343 2±0.001 5(~)	0.333 1±0.000 1(+)	0.289 8±0.045 9(+)
KUR	0.989 6±0.000 9	0.974 4±0.000 5(+)	0.978 8±0.013 4(~)	0.976 7±0.015 4(~)	0.298 9±0.000 1(+)	0.594 3±0.000 1(+)
ZDT1	0.996 7±0.000 0	0.536 4±0.000 2(+)	0.520 0±0.000 0(+)	0.940 7±0.000 0(+)	0.991 5±0.001 0(~)	0.4578±0.0317(+)
ZDT2	0.999 4 ±0.000 0	0.261 9±0.037 8(+)	0.462 5±0.026 6(+)	0.348 9±0.024 4(+)	0.937 5±0.000 1(+)	0.7101±0.0156(+)
ZDT3	0.953 2±0.009 9	0.692 1±0.007 1(+)	0.427 4±0.000 8(+)	0.759 5±0.016 4(+)	0.950 2±0.000 3(~)	0.4508±0.0358(+)
ZDT4	0.998 5±0.000 0	0.431 1±0.031 5(+)	0.058 8±0.094 4(+)	0.296 2±0.073 8(+)	0.718 0±0.000 2(+)	0.2613±0.0112(+)
ZDT6	0.999 9±0.000 0	0.007 1±0.000 6(+)	0.927 9±0.000 0(+)	0.309 4±0.000 7(+)	0.999 9±0.000 0(~)	0.9758±0.0003(+)
DTLZ2	0.987 1±0.000 1	0.926 9±0.000 5(+)	0.922 9±0.054 1(~)	0.987 4±0.054 4(~)	0.841 2±0.000 0(+)	0.5896±0.0151(+)
DTLZ4	1.000 0±0.006 9	0.996 7±0.030 3(~)	0.494 7±0.000 0(+)	0.959 1±0.000 0(+)	0.999 4±0.000 0(~)	0.9980±0.0064(~)
DTLZ5	0.997 0±0.000 0	0.619 5±0.000 0(+)	0.658 5±0.000 0(+)	0.691 6±0.000 0(+)	0.948 4±0.000 0(+)	0.673 2±0.253 2(+)
DTLZ6	0.982 0±0.000 0	0.680 6±0.009 6(+)	0.693 8±0.001 8(+)	0.702 6±0.064 0(+)	0.899 1±0.001 2(+)	0.471 6±0.038 4(+)
DTLZ7	0.998 8±0.000 3	0.892 4±0.003 3(+)	0.778 6±0.001 4(+)	0.905 5±0.025 8(+)	0.989 5±0.001 3(~)	0.709 1±0.005 7(+)

为直观地展示所提出的算法及比较算法所获得非支配解的质量和分布情况, 选取各算法 30 次运行的最好结果绘制非支配解集的分布图, 图 9, 图 10 及图 11 给出了 ZDT1, ZDT2 及 ZDT3 测试函数的非支配解集的分布图. 通过对以上指标的数据分析及分布图可以发现, 所提出的算法能够用较少的估值次数, 获得具有较好质量和分布性的非支配解集.

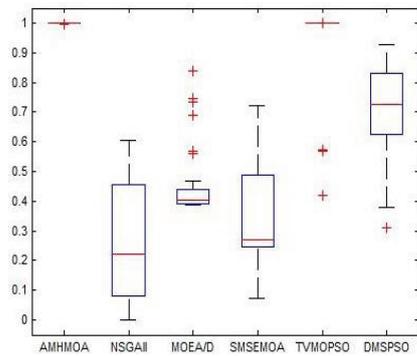


图7 ZDT2 测试函数的30次运行 Hypervolume 指标实验结果的盒状图

Fig. 7 Boxplot of experimental result via Hypervolume-metric by 30 runs for ZDT2

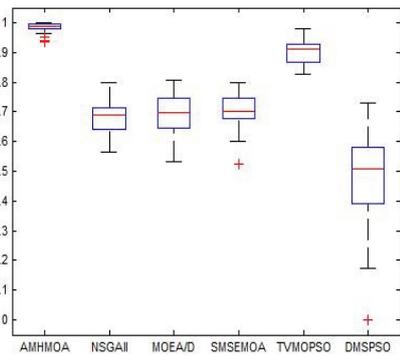


图8 DTLZ6 测试函数的30次运行 Hypervolume 指标实验结果的盒状图

Fig. 8 Boxplot of experimental result via Hypervolume-metric by 30 runs for DTLZ6

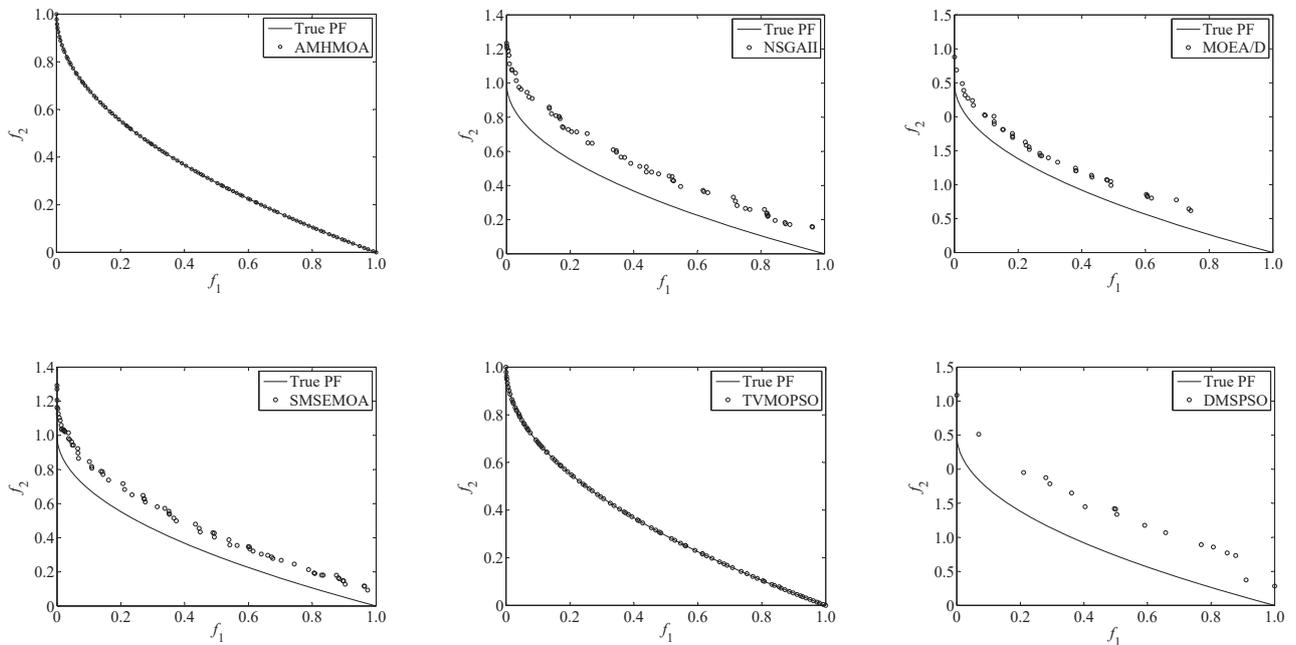


图9 六种算法获得的 ZDT1 的非支配解集分布图

Fig. 9 The distribution of nondominant solution sets of ZDT1 obtained by six algorithms

4 结束语

本文提出了一种基于自适应多种群策略的混合多目标优化算法. 该算法将多目标优化问题分解为多个单目标子问题, 首先引入部分均匀分布的子问题, 以快速地获得这些子问题的近似最优解, 然后在这些优势解的基础上逐步引入其他的子问题进行求解. 算法在每次迭代时, 根据种群在目标空间和解空间的分布情况为当前的子问题构造子种群, 并采用粒子群优化算法对子问题最优解进行搜索, 采用差分进化算法对外部档案实施优化. 种群和外部档案之间实施信息交互策略加快了算法的收敛速度. 通过对标准测试函数进行仿真实验, 并与五个典型的多目标优化算法进行比较, 结果表明所提出的算法能够利用较少的估值次数, 获得具有较好质量和分布性的非支配解集. 当然, 还有许多工作需要将来继续深入研究, 如与更多算法进行

比较以及选择不同的测试问题, 以检验所提出算法在求解多目标优化问题的性能; 寻找一种有效的策略以平衡种群和外部档案进化的计算资源分配; 对子种群规模等参数进行灵敏性分析也是非常必要的.

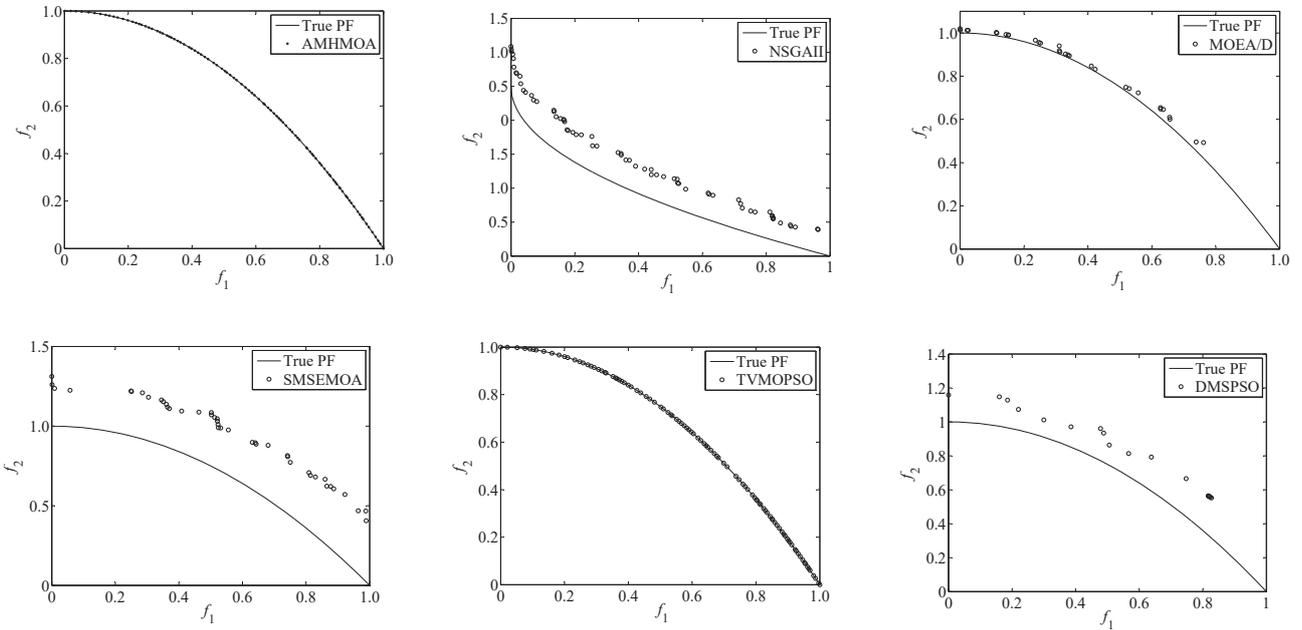


图 10 六种算法获得的 ZDT2 的非支配解集分布图

Fig. 10 The distribution of nondominant solution sets of ZDT2 obtained by six algorithms

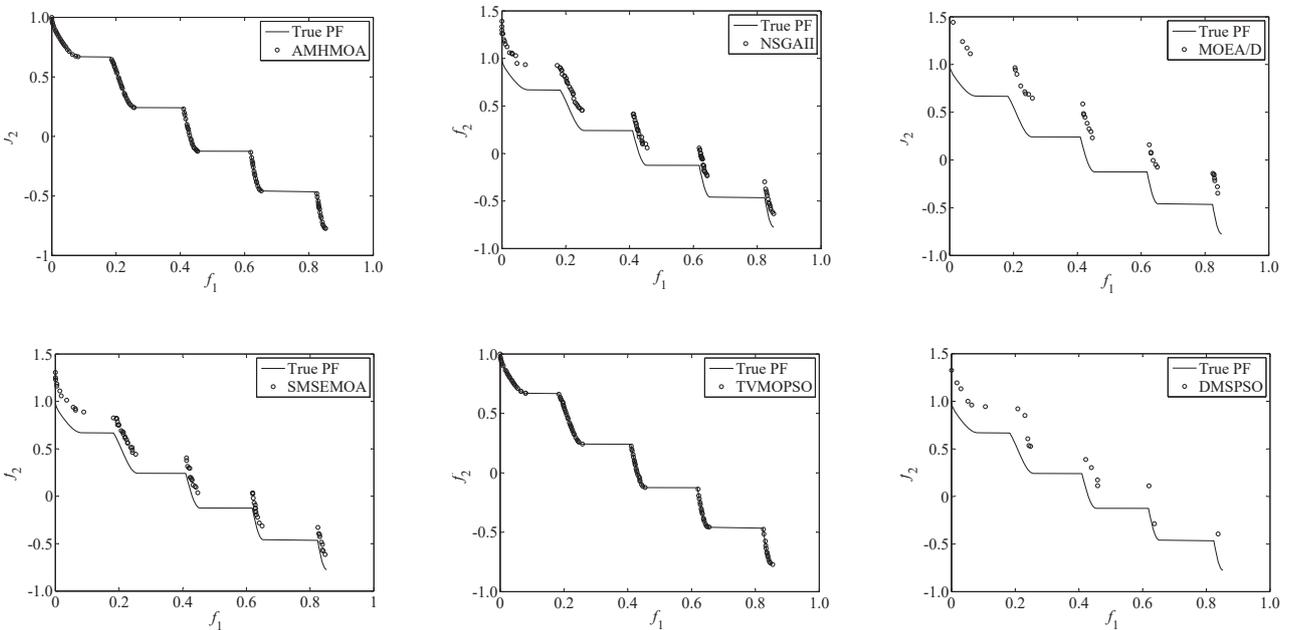


图 11 六种算法获得的 ZDT3 的非支配解集分布图

Fig. 11 The distribution of nondominant solution sets of ZDT3 obtained by six algorithms

参考文献:

[1] Burke E K, Kendall G. Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques. Second Edition, Berlin: Springer, 2014: 403-449.

- [2] Deb K, Pratap A, Agarwal S, et al. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 2002, 6(2): 182–197.
- [3] Tang L, Wang X. A hybrid multiobjective evolutionary algorithm for multiobjective optimization problems. *IEEE Transactions on Evolutionary Computation*, 2013, 17(1): 20–45.
- [4] Ali M, Siarry P, Pant M. An efficient differential evolution based algorithm for solving multi-objective optimization problems. *European Journal of Operational Research*, 2012, 217(2): 404–416.
- [5] Wang Y N, Wu L H, Yuan X F. Multi-objective self-adaptive differential evolution with elitist archive and crowding entropy-based diversity measure. *Soft Computing*, 2010, 14(3): 193–209.
- [6] Zitzler E, Künzli S. Indicator-based selection in multiobjective search // *Proceedings of Eighth International Conference on Parallel Problem Solving from Nature-PPSN VIII*. Berlin: Springer-Verlag, 2004: 832–842.
- [7] Beume N, Naujoks B, Emmerich M. SMS-EMOA: Multiobjective selection based on dominated hypervolume. *European Journal of Operational Research*, 2007, 181(3): 1653–1669.
- [8] Zhang Q, Li H. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation*, 2007, 11(6): 712–731.
- [9] Ma X, Liu F, Qi Y, et al. MOEA/D with opposition-based learning for multiobjective optimization problem. *Neurocomputing*, 2014, 146(1): 48–64.
- [10] Tan Y, Jiao Y, Li H, et al. MOEA/D+ uniform design: A new version of MOEA/D for optimization problems with many objectives. *Computers & Operations Research*, 2013, 40(6): 1648–1660.
- [11] Qi Y, Ma X, Liu F, et al. Moea/d with adaptive weight adjustment. *Evolutionary Computation*, 2014, 22(2): 231–264.
- [12] 王宇平, 焦永昌, 张福顺. 解多目标优化的均匀正交遗传算法. *系统工程学报*, 2004, 18(6): 481–486.
Wang Y P, Jiao Y C, Zhang F S. Uniform and orthogonal genetic algorithm for multiobjective optimization. *Journal of Systems Engineering*, 2004, 18(6): 481–486. (in Chinese)
- [13] Jaimes A L, Coello C A. MRMOGA: A new parallel multi-objective evolutionary algorithm based on the use of multiple resolutions. *Concurrency and Computation: Practice and Experience*, 2007, 19(4): 397–441.
- [14] Zhang Y, Gong D, Ding Z. Handling multi-objective optimization problems with a multi-swarm cooperative particle swarm optimizer. *Expert Systems with Applications*, 2011, 38(11): 13933–13941.
- [15] Tan K C, Yang Y J, Goh C K. A distributed cooperative coevolutionary algorithm for multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 2006, 10(5): 527–549.
- [16] Jiao L C, Wang H, Shang R H, et al. A co-evolutionary multi-objective optimization algorithm based on direction vectors. *Information Sciences*, 2013, 228(7): 90–112.
- [17] Liang J J, Qu B Y, Suganthan P N, et al. Dynamic multi-swarm particle swarm optimization for multi-objective optimization problems // *Proceedings of the 2012 IEEE Congress on Evolutionary Computation*. Brisbane: IEEE Press, 2012: 1–8.
- [18] Miettinen K. *Nonlinear Multiobjective Optimization*. Norwell: Kluwer, 1999.
- [19] Parrott D, Li X. Locating and tracking multiple dynamic optima by a particle swarm model using speciation. *IEEE Transactions on Evolutionary Computation*, 2006, 10(4): 440–458.
- [20] Coello C A C, Pulido G T, Lechuga M S. Handling multiple objectives with particle swarm optimization. *IEEE Transactions on Evolutionary Computation*, 2004, 8(3): 256–279.
- [21] Tripathi P K, Bandyopadhyay S, Pal S K. Multi-objective particle swarm optimization with time variant inertia and acceleration coefficients. *Information Sciences*, 2007, 177(22): 5033–5049.
- [22] Huang V L, Suganthan P N, Liang J J. Comprehensive learning particle swarm optimizer for solving multiobjective optimization problems. *International Journal of Intelligent Systems*, 2006, 21(2): 209–226.
- [23] Esquivel S C, Coello Coello C A. On the use of particle swarm optimization with multimodal functions // *Proceeding of IEEE International Conference on Evolutionary Computation*. Canberra: IEEE Press, 2003, 2: 1130–1136.
- [24] Huband S, Hingston P, Barone L, et al. A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Transactions on Evolutionary Computation*, 2006, 10(5): 477–506.
- [25] Schaffer J D. Multiple objective optimization with vector evaluated genetic algorithms // *Proceedings of the First International Conference on Genetic Algorithms*. Grefenstette J J. Hillsdale, NJ: Lawrence Erlbaum, 1985: 93–100.