

多种群约束多目标优化算法的资源分配指标设计

方静静

广东工业大学数学与统计学院, 广东 广州

收稿日期: 2023年3月6日; 录用日期: 2023年4月16日; 发布日期: 2023年4月23日

摘要

本文研究基于多种群的约束多目标优化算法的种群间资源分配问题。本文提出了一个新的指标来衡量种群资源的合理分配。所提出的指标根据不同种群的变化及进化代数, 为不同种群分配进化资源。它根据种群中理想点和最差点的变化, 来衡量该种群的变化。所提出的指标被嵌入到一个先进的多种群约束多目标算法。在实验中, 通过在20个基准函数的数据实验, 本文展示了所提出指标可以有效地为不同种群分配资源。

关键词

约束多目标, 优化, 多种群, 资源分配

Resource Allocation Indicator Design of Multi-Population-Based Constrained Multi-Objective Optimization Algorithm

Jingjing Fang

School of Mathematics and Statistics, Guangdong University of Technology, Guangzhou Guangdong

Received: Mar. 6th, 2023; accepted: Apr. 16th, 2023; published: Apr. 23rd, 2023

Abstract

In this paper, we study the problem of resource allocation among multiple populations on multi-populations-based constrained multi-objective optimization algorithm. This paper proposes a new indicator to measure the allocation of population evolutionary resources. The proposed in-

indicator allocates evolutionary resources for different populations according to the changes of different populations and evolutionary generations. It measures the change of the population according to the change of the ideal point and the nadir point in the population. The proposed indicator is embedded into an advanced multi-population constrained multi-objective algorithm. In the numerical experiment, this paper shows that the proposed indicators can effectively allocate resources for different populations on 20 benchmark functions.

Keywords

Constrained Multi-Objective, Optimization, Multi-Population, Resource Allocation

Copyright © 2023 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

进化约束多目标优化算法被认为是处理约束多目标优化问题的有效方法[1]。近年来,许多进化约束多目标优化算法被提出并被运用到工业和学术界等[2][3]。其中,使用多个种群的混合算法展现出了显著的效果[4]。例如,C-TAEA [5]使用了两个种群协同探索,一个种群以收敛性为指导进行搜索,使得种群逼近 Pareto 前沿,另一个种群以多样性为指导搜索未被第一个种群搜索过的空间,以此来保证多样性。新个体由两个种群杂交变异产生。ICMA [6]提出了一个新的指标来衡量个体对探索有前景区域的贡献值。在 ICMA 中,一个工作种群通过保存贡献值最大的一些个体来尽可能地均匀探索整个目标空间,另一个归档种群通过保存搜索到的精英有效解进行归档。新个体在进化前期更多的由工作种群产生,而后期则更多的由归档种群产生。CCMO [7]提出一种协同进化框架,一个种群以求解原问题为目标进行优化,另一个种群以求解从原问题提取出来的协助问题为目标进行优化。两个种群分开独立产生个体,并在种群更新时交换信息。

尽管先前的基于多种群的混合算法在求解约束多目标优化问题时展示出了强大的效果,但种群间资源的分配只是简单的平分或者根据进化代数变化。在进化过程中根据不同的优化问题的特征进行种群间资源的合理分配问题仍鲜有研究。然而,合理地分配种群间的资源,可以有效地提高算法的效率。

为了处理多种群间资源的分配问题,本文提出了一种基于种群变化的指标来识别优化问题的特征,从而动态地分配资源。在进化过程中,种群的变化由一定代数的理想点和最差点的变化来衡量。所提出的指标根据种群间的变化的差异,在不同进化阶段以不同程度的更多的资源投入到变化较大的种群中。本文的主要贡献包括以下几点:

- 1) 计算每个种群的理想点和最差点的变化,以它们的差异来识别问题的特征,从而更好地指导不同问题不同种群需要的资源;
- 2) 根据不同种群的差异,以及当前进化的阶段,设计一个新的指标。在一个先进的多种群混合优化算法上根据指标实现动态的种群间资源分配;
- 3) 广泛的实验验证了所提指标的有效性。

本文的其余部分如下。第二节介绍了所提出的指标。第三节介绍了实验和结果,并对其进行了分析。第四节为本文进行了总结。

2. 所提的框架

2.1. 种群变化的衡量

本文使用理想点和最差点的变化来衡量种群变化。不失一般性，本文考虑以下约束多目标：

$$\begin{cases} \text{minimize} & \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), \dots, f_m(\mathbf{x}))^T \\ \text{subject to} & g_i(\mathbf{x}) \geq 0, i = 1, \dots, q \\ & h_j(\mathbf{x}) = 0, j = 1, \dots, p \\ & \mathbf{x} \in \mathbb{R}^n \end{cases} \quad (1)$$

则第 t 代种群 $P^t = \{x_1^t, x_2^t, \dots, x_N^t\}$ 中，理想点 $Z^t = \{z_1^t, z_2^t, \dots, z_m^t\}$ 和最差点 $n^t = \{n_1^t, n_2^t, \dots, n_m^t\}$ 的定义分别为：

$$z_i^t = \min_{j=1,2,\dots,N} f_i(x_j) \quad (2)$$

$$n_i^t = \max_{j=1,2,\dots,N} f_i(x_j) \quad (3)$$

在第 t 代，理想点的变化 cz_t 和最差点的变化 cn_t 由当前进化代数 t 与最近的 k 代分别定义：

$$cz_t = \max_{i=1,2,\dots,m} \left\{ \left| z_i^t - z_i^{t-k} \right| / \left| z_i^{t-k} \right| \right\} \quad (4)$$

$$cn_t = \max_{i=1,2,\dots,m} \left\{ \left| n_i^t - n_i^{t-k} \right| / \left| n_i^{t-k} \right| \right\} \quad (5)$$

种群的变化由理想点的变化和最差点的变化定义为：

$$cp_t = \max \{cz_t, cn_t\} \quad (6)$$

2.2. 所提的资源分配指标

假设两个种群 P_1 和 P_2 ，本文使用上述种群变化的衡量两个种群在第 t 代的变化分别为 cp_1^t 和 cp_2^t 。在进化第 t 代，本文提出的基于种群变化的资源分配比例为：

$$R = (1 + 1/cp_1^t) / (1 + 1/cp_2^t) \times t / T_{\max} \quad (7)$$

其中 T_{\max} 为最大进化代数。根据资源分配比例，种群 P_1 和 P_2 的后代产生个数分别为：

$$N_1 = R \times N \quad (8)$$

$$N_2 = (1 - R) \times N \quad (9)$$

所提出的资源分配的指标具有以下两个特点：

- 1) 当种群 P_1 的比例变化比 P_2 较大时，更多的资源倾向于分配给 P_1 ，即 P_1 产生并评估更多的后代；反之，则 P_2 产生并评估更多的后代。
- 2) 进化的前期更倾向于分配更多的资源给 P_2 ，进化的后期更偏好于将更多的资源分配给 P_1 。

2.3. 所提的算法

本文将所提的资源分配指标运用于双种群约束多目标优化算法 CCMO 上，动态地分配基于以求解原问题为目标的种群的资源和以求解从原问题提取出来的协助问题为目标的种群的资源。我们令以求解原问题为目标的种群为 P_1 ，令以求解从原问题提取出来的协助问题为目标的种群为 P_2 ，即进化前期更倾向于保证算法的收敛性，而进化的后期更强调种群的可行性和多样性。算法 1 为所提算法的框架。在第 t

代, 如果 t 大于等于 k , 则开始计算种群的变化, 并根据种群的变化和当前所在代数, 即公式(8)和(9)计算每个种群所产生的后代数目。在选择阶段则通过从所产生的所有后代及种群选择个体更新种群。最后输出以求解原问题为目标的最终种群 P_1 。

算法 1. 所提算法的框架

输入: 种群 P_1 , 种群 P_2 , 种群数目 N , 最大进化代数 T_{\max} , 相隔代数 k

输出: 最终种群 P_1

1. 初始化: $t = 0$
2. 当 $t \leq T_{\max}$ 时, 执行以下:
3. 如果 $t < k$, 则 $N_1 = N_2 = N/2$;
4. 如果 $t \geq k$, 则根据(8)和(9)分别计算 N_1 和 N_2 ;
5. 由 P_1 产生 N_1 个后代 Off_1 ;
6. 由 P_2 产生 N_2 个后代 Off_2 ;
7. 从 $P_1 \cup Off_1 \cup Off_2$ 根据 P_1 的目标选择 N 个个体组成 P_1 ;
8. 从 $P_2 \cup Off_1 \cup Off_2$ 根据 P_2 的目标选择 N 个个体组成 P_2 ;
9. 结束循环
10. 输出最终种群 P_1

3. 数据实验与分析

3.1. 参数设置

- 1) 种群规模 $N = 100$;
- 2) 最大进化代数 $T_{\max} = 500$, 相隔代数 $k = 5$;
- 3) 统计检验: 我们使用显著性水平为 0.05 的 Wilcoxon 秩和检验来对算法进行比较, 符号“-”、“=”和“+”表示所提算法优于、相似和低于其他算法, 表格中突出显示的为每行的最佳结果;
- 4) 对比算法的参数设置与它们各自的原文保持一致。

3.2. 基准函数和对比算法

本文选用 DC_DTLZ [5] 系列和 MW [8] 系列作为约束多目标优化问题的基准测试函数。另外, 本文选择了 4 种目前先进的 CMOEA, 即 TiGE₂ [9]、ToP [10]、I-DBEA [11] 和 CCMO 进行性能比较。对比算法的主要思想如下:

1) TiGE₂: 它旨在平衡种群的多样性、收敛性和可行性。因此它分别为这三种性能设计了三个指标。基于设计的三个指标, 提出了一种用于求解 CMOP 三目标进化框架。

2) ToP: 它是一种简单的两阶段约束多目标进化算法。第一阶段用于寻找有希望的可行区域; 第二阶段使用现有的 CMOEA 迫使种群收敛到可行区域的帕累托前沿界面。它在处理目标空间和决策空间都有约束的优化问题上表现良好。

3) I-DBEA: 它是一个可行驱动的约束多目标进化算法。它通过参数 ε 来扩大可行解的范围, 即违反约束程度小于 ε 的个体也被视为可行解。因此它可以充分利用不可行解携带的有效信息, 帮助种群穿越不可行带, 从而加速算法的收敛。

4) CCMO: 它提出一种双种群的协同进化框架。它将无约束的优化问题作为辅助问题, 来帮助解决原始问题。两个种群可以信息共享, 共同促进种群的进化。

3.3. 算法性能度量指标

反转世代距离(Inverted Generational Distance, IGD) [12]和超体积(Hypervolume, HV) [13]是两种常见的算法性能评价指标。本文选用这两个指标来评估约束优化算法的性能。IGD 和 HV 的具体介绍分别如下:

1) IGD: 设 P^* 是一组已知, 并且均匀分布在 PF 上的点集, X 是使用算法所求得的最优解集。则 X 的 IGD 值可以由以下公式计算:

$$\text{IGD}(P^*, X) = \frac{\sum_{v \in P^*} d(v, X)}{|P^*|} \quad (10)$$

其中 $d(v, X)$ 表示点 v 到集合 X 的最小距离, $|P^*|$ 是集合 P^* 的规模。IGD 值越小, 说明算法的性能越好。

2) HV: HV 测量的是可行解集 X 和参考点 z 所围成区域的超体积。HV 值可以由下列公式计算:

$$\text{HV}(X | z) = \text{VOL} \left(\bigcup_{x \in X} [f_1(x), z_1] \times \cdots \times [f_m(x), z_m] \right) \quad (11)$$

其中 $\text{VOL}(\ast)$ 是用于测量体积的度量指标。参考点 z 一般由每个目标上的最大函数值的 1.1 倍组成, 因此 PF 上的任意一个可行解都可以支配参考点 z 。HV 值越大, 说明算法的性能越好。

3.4. 所提的算法表现分析

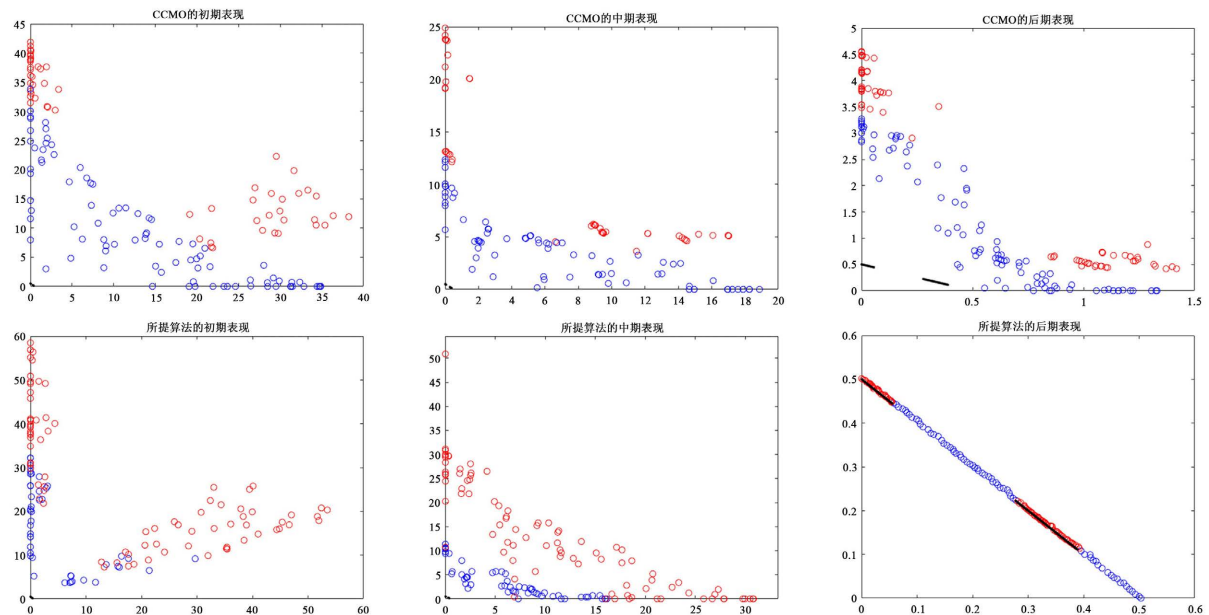


Figure 1. Initial, intermediate and late performance of CCMO and proposed algorithm on DC3_DTLZ1

图 1. CCMO 和所提算法在 DC3_DTLZ1 上的初、中和后期表现

其中, 红色种群为以求解原问题为目标的种群, 蓝色种群为以求解从原问题提取出来的协助问题为目标的种群, 黑色边界为真实 PF。

本节在一个测试问题 DC3_DTLZ1 上展示所提的算法的表现。图 1 展示了 CCMO 和所提的算法在 DC3_DTLZ1 上初期, 中期和后期的种群分布。从图 1 可以看出, 所提算法具有更高的效率探索到真实的 PF。在初期, 种群距离 PF 非常远, 此时以求解从原问题提取出来的协助问题为目标的种群(蓝色种群)

快速逼近 PF，从而更多的资源分配给它，此时根据我们提出的指标得到红色种群和蓝色种群的资源分配比为 1:9，算法更加考虑收敛性。到了进化的中期，尽管种群距离 PF 仍然比较远，但是此时以求解从原问题提取出来的协助问题为目标的种群(蓝色种群)变化的速率下降，根据所提的指标，此时红色种群和蓝色种群的资源分配为 3:7，算法同时考虑收敛性和可行性。到了进化的后期，两个种群都已经非常接近 PF，此时所提的指标将更多的资源放在了以求解原问题为目标的种群(红色种群)上，红色种群和蓝色种群的资源分配为 7:3。到了进化的后期或者两个种群都很接近各自的 PF 时，所提算法更强调在以求解原问题为目标的种群上分配资源，从而保证种群的可行性和多样性。

3.5. 所提的算法在基准函数上的表现

Table 1. IGD average value of the five algorithms on test instances

表 1. 5 个算法在测试实例上的 IGD 平均值

Problem	M	D	TiGE ₂	ToP	I-DBEA	CCMO	Ours
DC1_DTLZ1	2	6	7.7155e-1 -	1.6343e-1 -	1.0130e-2 +	5.3716e-1 -	7.8894e-2
DC1_DTLZ3	2	11	2.3128e+0 -	1.1296e+0 -	3.7545e-1 -	1.1733e+0 -	3.4909e-1
DC2_DTLZ1	2	6	1.4080e-1 -	NaN -	3.6107e-1 -	8.6613e-2 =	3.9525e-2
DC2_DTLZ3	2	11	5.4186e-1 -	NaN -	9.3103e-1 =	5.0134e-1 -	2.5436e-1
DC3_DTLZ1	2	6	2.3101e+0 -	2.8659e+0 -	1.0849e+0 -	1.2739e+0 -	5.3395e-1
DC3_DTLZ3	2	11	3.7561e+0 -	3.6862e+0 -	2.4646e+0 -	2.0394e+0 -	7.1301e-1
MW1	2	15	1.0984e-1 =	NaN -	4.8522e-1 -	1.1099e-1 =	1.4445e-1
MW2	2	15	4.0653e-2 +	1.9326e-1 =	5.1052e-1 -	1.0975e-1 =	1.2987e-1
MW3	2	15	3.3796e-2 -	6.7104e-1 -	5.0507e-1 -	5.6928e-3 -	5.5625e-3
MW4	3	15	8.4430e-2 =	NaN -	5.2377e-1 -	9.1248e-2 =	9.8783e-2
MW5	2	15	6.8080e-2 =	NaN -	6.5947e-1 -	3.1712e-1 =	3.2222e-1
MW6	2	15	1.8588e-1 +	1.1014e+0 -	7.7884e-1 -	5.1590e-1 =	4.8563e-1
MW7	2	15	4.1996e-2 -	4.8592e-1 -	6.3381e-1 -	6.9709e-3 =	6.7480e-3
MW8	3	15	1.0412e-1 +	9.1731e-1 -	7.9012e-1 -	1.3487e-1 =	1.4839e-1
MW9	2	15	1.4363e-1 =	NaN -	6.5638e-1 -	5.9807e-1 -	4.0667e-1
MW10	2	15	7.9026e-2 +	NaN -	5.3594e-1 -	4.7687e-1 =	3.7931e-1
MW11	2	15	5.0902e-2 -	8.9601e-1 -	1.0904e+0 -	8.5050e-3 -	8.1891e-3
MW12	2	15	1.5637e-1 =	NaN -	7.3341e-1 -	4.4985e-1 =	4.3089e-1
MW13	2	15	4.0789e-1 -	8.4542e-1 -	1.2650e+0 -	2.3922e-1 =	2.6381e-1
MW14	3	15	1.6586e-1 =	3.6381e-1 -	2.7237e+0 -	2.3159e-1 -	1.9394e-1
			+/-/=				
			4/10/6	0/19/1	1/18/1	0/9/11	

本节将所提的算法与四个对比算法在基准函数上进行比较。每个算法在基准函数的每个测试问题上独立运行三十遍。表 1 和表 2 分别展示了五个算法在每个测试问题上的所求 IGD 和 HV 的平均值。其中，若三十次独立实验都没有求解到可行解，则记为“NaN”。同时，我们对每个对比算法在每个测试问题上与所求算法进行了统计检验。从表 1 和表 2 可以看到，所提算法在 20 个测试问题上明显优于原双种群约束多目标优化算法 CCMO，在 IGD 和 HV 上分别都有 9 个更优的表现。这个结果是符合我们所期望的，

因为所提的算法与 CCMO 使用相同的选择算子，并在种群的资源分配上进行了优化。这使得所提的算法能够更有效的求解约束多目标优化问题。

另外，在 IGD 上，所提的算法在 20 个测试问题上分别在 10 个、19 个和 18 个上优于对比算法 TiGE₂、ToP 和 I-DBEA。在 HV 上，所提的算法在 20 个测试问题上分别在 8 个、20 个和 18 个上优于对比算法 TiGE₂、ToP 和 I-DBEA。这是因为所提算法保存了 CCMO 算法的优势，同时，资源的更优分配保证了所提的算法的有效性。

Table 2. HV average value of the five algorithms on test instances
表 2. 5 个算法在测试实例上的 HV 平均值

Problem	M	D	TiGE ₂	ToP	I-DBEA	CCMO	Ours
DC1_DTLZ1	2	6	1.5709e-2 -	2.2384e-1 -	3.5277e-1 =	1.0439e-1 -	2.8884e-1
DC1_DTLZ3	2	11	0.0000e+0 -	7.7176e-2 -	5.5383e-2 -	6.9636e-2 -	1.6019e-1
DC2_DTLZ1	2	6	3.1637e-1 -	NaN -	2.5231e-2 -	4.4253e-1 =	5.2832e-1
DC2_DTLZ3	2	11	4.4556e-2 =	NaN -	0.0000e+0 =	5.0979e-2 -	1.9041e-1
DC3_DTLZ1	2	6	0.0000e+0 -	0.0000e+0 -	3.9645e-3 -	1.5366e-2 -	9.0473e-2
DC3_DTLZ3	2	11	0.0000e+0 -	5.9961e-3 -	0.0000e+0 -	2.5913e-2 -	9.2890e-2
MW1	2	15	3.8632e-1 =	NaN -	9.6199e-2 -	4.1374e-1 =	3.7017e-1
MW2	2	15	5.2593e-1 +	3.5139e-1 -	1.4687e-1 -	4.3420e-1 =	4.1505e-1
MW3	2	15	5.2146e-1 -	1.0354e-1 -	1.5521e-1 -	5.4249e-1 -	5.4272e-1
MW4	3	15	7.9366e-1 =	NaN -	2.3766e-1 -	7.6632e-1 =	7.5359e-1
MW5	2	15	2.8085e-1 =	NaN -	3.1437e-2 -	2.0862e-1 =	2.0739e-1
MW6	2	15	2.4007e-1 +	1.3206e-2 -	4.9488e-2 -	1.0682e-1 =	1.1417e-1
MW7	2	15	3.8678e-1 -	1.7493e-1 -	1.2718e-1 -	4.1113e-1 =	4.1128e-1
MW8	3	15	4.7416e-1 +	5.6806e-2 -	5.3095e-2 -	3.5945e-1 =	3.3693e-1
MW9	2	15	2.9145e-1 =	NaN -	6.1033e-2 -	6.4076e-2 -	1.5392e-1
MW10	2	15	3.8528e-1 +	NaN -	1.4532e-1 -	1.9066e-1 =	2.3611e-1
MW11	2	15	4.2822e-1 -	2.1954e-1 -	1.5354e-1 -	4.4542e-1 -	4.4580e-1
MW12	2	15	4.7184e-1 =	NaN -	9.5133e-2 -	2.6877e-1 =	2.5174e-1
MW13	2	15	3.4091e-1 =	2.1107e-1 -	1.8836e-1 -	3.3801e-1 =	3.3237e-1
MW14	3	15	4.4621e-1 +	3.8220e-1 -	1.0638e-2 -	4.1791e-1 -	4.2966e-1
+/-/=			5/8/7	0/20/0	0/18/2	0/9/11	

4. 总结

考虑到基于多种群的约束多目标优化算法在进化过程中不同种群的变化不同，并且指导其各自进化的原则不同，需要在不同进化阶段对不同种群进行合理的资源分配。本文通过每个种群的变化及当前所在的进化代数，设计了资源分配的指标。根据所提的指标，每个种群产生不同数目的后代。另外，本文将所提的指标运用于一个双种群的约束多目标优化算法上。数据实验结果显示了所提的算法能够更有效地分配种群的资源，从而在不同的测试问题上表现优于四个现有的先进的约束多目标进化算法。

参考文献

- [1] Liang, J., Ban, X.X., Yu, K.J., *et al.* (2022) A Survey on Evolutionary Constrained Multiobjective Optimization. *IEEE Transactions on Evolutionary Computation*, **27**, 201-221. <https://doi.org/10.1109/TEVC.2022.3155533>
- [2] Fan, Z., Li, W., Cai, X., *et al.* (2019) An Improved Epsilon Constraint-Handling Method in MOEA/D for CMOPs with Large Infeasible Regions. *Soft Computing*, **23**, 12491-12510. <https://doi.org/10.1007/s00500-019-03794-x>
- [3] Fan, Z., Li, W., Cai, X., *et al.* (2019) Push and Pull Search for Solving Constrained Multi-Objective Optimization Problems. *Swarm and Evolutionary Computation*, **44**, 665-679. <https://doi.org/10.1016/j.swevo.2018.08.017>
- [4] Wang, J., Liang, G. and Zhang, J. (2018) Cooperative Differential Evolution Framework for Constrained Multiobjective Optimization. *IEEE Transactions on Cybernetics*, **49**, 2060-2072. <https://doi.org/10.1109/TCYB.2018.2819208>
- [5] Li, K., Chen, R., Fu, G., *et al.* (2018) Two-Archive Evolutionary Algorithm for Constrained Multiobjective Optimization. *IEEE Transactions on Evolutionary Computation*, **23**, 303-315. <https://doi.org/10.1109/TEVC.2018.2855411>
- [6] Yuan, J., Liu, H.L., Ong, Y.S., *et al.* (2021) Indicator-Based Evolutionary Algorithm for Solving Constrained Multiobjective Optimization Problems. *IEEE Transactions on Evolutionary Computation*, **26**, 379-391. <https://doi.org/10.1109/TEVC.2021.3089155>
- [7] Tian, Y., Zhang, T., Xiao, J., *et al.* (2020) A Coevolutionary Framework for Constrained Multiobjective Optimization Problems. *IEEE Transactions on Evolutionary Computation*, **25**, 102-116. <https://doi.org/10.1109/TEVC.2020.3004012>
- [8] Asafuddoula, M., Ray, T. and Sarker, R. (2015) A Decomposition-Based Evolutionary Algorithm for Many Objective Optimization. *IEEE Transactions on Evolutionary Computation*, **19**, 445-460. <https://doi.org/10.1109/TEVC.2014.2339823>
- [9] Zhou, Y., Zhu, M., Wang, J., *et al.* (2018) Tri-Goal Evolution Framework for Constrained Many-Objective Optimization. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, **50**, 3086-3099. <https://doi.org/10.1109/TSMC.2018.2858843>
- [10] Liu, Z.Z. and Wang, Y. (2019) Handling Constrained Multiobjective Optimization Problems with Constraints in Both the Decision and Objective Spaces. *IEEE Transactions on Evolutionary Computation*, **23**, 870-884. <https://doi.org/10.1109/TEVC.2019.2894743>
- [11] Ma, Z. and Wang, Y. (2019) Evolutionary Constrained Multiobjective Optimization: Test Suite Construction and Performance Comparisons. *IEEE Transactions on Evolutionary Computation*, **23**, 972-986. <https://doi.org/10.1109/TEVC.2019.2896967>
- [12] Bosman, P.A.N. and Thierens, D. (2003) The Balance between Proximity and Diversity in Multiobjective Evolutionary Algorithms. *IEEE Transactions on Evolutionary Computation*, **7**, 174-188. <https://doi.org/10.1109/TEVC.2003.810761>
- [13] Zitzler, E. and Thiele, L. (1998) Multiobjective Optimization Using Evolutionary Algorithms—A Comparative Case Study. *Parallel Problem Solving from Nature—PPSN V: 5th International Conference*, Amsterdam, 27-30 September 1998, 292-301. <https://doi.org/10.1007/BFb0056872>