

基于全局边际排序和自适应参数调节的多目标粒子群算法

唐湘黔

贵州大学数学与统计学院, 贵州 贵阳

收稿日期: 2022年11月14日; 录用日期: 2022年12月8日; 发布日期: 2022年12月15日

摘要

多目标粒子群优化算法作为一种强大的优化技术, 在各个领域得到了广泛的应用。然而, 对于大多数现有的算法来说, 同时在收敛性和多样性方面表现良好仍然是一项具有挑战性的任务。针对上述挑战, 本文提出了一种基于全局边际排序和自适应参数调节的多目标粒子群优化算法(GGRAW)。在多目标粒子群算法中, 飞行参数、惯性权重 ω 、加速度系数 c_1 和 c_2 对于实现种群开发和勘探之间的平衡非常重要。该策略利用个体在客观空间中的位置信息, 以获得整个种群的优势边际。该方法不仅考虑了种群的分布, 还考虑了个体的相关信息。通过自适应参数的调节更加促使种群中的粒子向真实的Pareto前沿移动; 将GGRAW算法与3种经典的多目标粒子群算法在ZDT、DTLZ和WFG系列的部分测试函数上进行仿真实验; 试验结果表明: 相对于其他的几种经典算法, GGRAW算法表现出较好的收敛性和多样性, 因此, GGRAW算法可以作为求解多目标优化问题的有效算法。

关键词

多目标优化, 粒子群算法, 全局边际排序, 自适应参数

Multi-Objective Particle Swarm Optimization Based on Global Marginal Ranking and Adaptive Parameter Adjustment

Xiangqian Tang

School of Mathematics and Statistics, Guizhou University, Guiyang Guizhou

Received: Nov. 14th, 2022; accepted: Dec. 8th, 2022; published: Dec. 15th, 2022

Abstract

As a powerful optimization technology, multi-objective particle swarm optimization algorithm has been widely used in various fields. However, performing well in both convergence and diversity remains a challenging task for most existing algorithms. To address these challenges, this paper proposes a multi-objective particle swarm optimization algorithm (GGRAW) based on global marginal sorting with adaptive parameter adjustment. In multi-objective particle swarm optimization, flight parameters, inertia weight ω , acceleration coefficients c_1 and c_2 are very important to achieve a balance between population exploitation and exploration. This strategy uses the position information of individuals in objective space to obtain the dominant margin of the whole population. This method not only considers the distribution of the population, but also considers the relevant information of the individual. By adjusting the adaptive parameters, the particles in the population can move towards the real Pareto frontier. The GGRAW algorithm and three classical multi-objective particle swarm optimization algorithms were simulated on some test functions of ZDT, DTLZ and WFG series. The experimental results show that compared with other classical algorithms, GGRAW algorithm shows better convergence and diversity. Therefore, GGRAW algorithm can be used as an effective algorithm for solving multi-objective optimization problems.

Keywords

Multi-Objective Optimization, Particle Swarm Optimization, Global Marginal Ranking, Adaptive Parameters

Copyright © 2022 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

在过去的十几年中,多目标优化受到了越来越多的关注,比如基于差分进化(DE) [1]、粒子群优化(PSO) [2]等遗传算法的自然启发元启发式算法。值得一提的是,基于种群的元启发式算法由于能够处理非线性、多模态和不连续的问题,已被广泛应用于求解真实的MOPs。其中,PSO是一种很有前途的群体智能技术,其灵感来自于鸟类和鱼类寻找食物的行为。由于其易于实现和快速收敛到全局最优,引起了研究者的极大兴趣,并广泛应用于许多单目标工程应用。

多目标优化问题(MOPs)的特点是多个目标同时优化[1]。由于MOP中的目标之间的冲突,几乎没有一个单一的解决方案能够同时满足所有的目标。因此,解决一个MOP的目标是找到一组帕累托非支配的解决方案,这需要在不同的目标之间进行权衡[3]。所有帕累托非支配解的集合称为帕累托最优集(PS),PS在目标空间中的映射称为帕累托前部(PF) [4]。当求解一个MOP时,在PS中得到的解不仅要尽可能地收敛于优化后的MOP的真实PF,而且要尽可能均匀地分布在PF上。一般来说,收敛性和多样性是评价最终解质量的两个关键标准。

为了应用PSO来解决MOP问题,我们必须解决以下两个基本问题。首先,如何选择全球最佳的解决方案(领导者)来指导群体搜索。由于没有单一存在所有目标中优于所有其他解决方案,因此许多非主导的解决方案可以被视为被选为领导者的候选方案。为了完成这项任务,人们必须采用一种合适的选择技

术,同时考虑到解决方案的多样性和收敛速度。然而在多目标优化过程中往往会忽略参数的设置,这可能会导致在优化过程中使得一些粒子陷入局部最优解。例如文献[5]中,对于参数的设置考虑欠妥,导致一些非劣解陷入局部最优,虽然在收敛性方面有所提高,但对于收敛性与多样性综合来看还是存在很大的不足。因此本文提出了基于全局边际排序和自适应参数调节的多目标粒子群算法(GGRAW)来进一步提高非劣解的多样性和收敛性。GGRAW 算法中加入了自适应参数的调节机制,更加有效地避免了种群陷入局部最优,大大提高了种群的收敛性和多样性。

2. 相关概念

2.1. 多目标优化问题

一般来说,多目标优化问题可以用数学方法描述为[6]

$$\begin{aligned} \min_{x \in \Omega} F(x) &= \{f_1(x), f_2(x), \dots, f_D(x)\} \\ \text{s.t. } g_j(x) &\leq 0, 0 \leq j \leq k \end{aligned}$$

其中 $x = (x_1, x_2, \dots, x_D)^T$ 是决策空间 Ω 中的 d 维决策向量, $g_j(x) \leq 0$ 表示 k 个等不等式约束, 映射函数 $f_i(x)$ 表示第 i 个目标函数, m 是目标函数的个数。

2.2. 粒子群优化算法

粒子群优化(PSO)是一种用于处理非线性优化问题的元启发式[6]。首先,初始化一组解决方案。然后,将种群经过多代人进行更新,以找到最优解。在演化过程中,每个粒子的位置通过两个学习算子[7]进行更新。一个是粒子在速度上的更新,另外一个是在位置上的更新。设 $x_i(t) = (x_{i,1}(t), x_{i,2}(t), \dots, x_{i,D}(t))$, $v_i(t) = (v_{i,1}(t), v_{i,2}(t), \dots, v_{i,D}(t))$ 分别为第 i 次迭代时粒子的位置和速度。然后,就可以用以下公式来更新该位置的每个维度:

$$v_{i,j}(t+1) = \omega v_{i,j}(t) + c_1 r_1 (pb_{i,j}(t) - x_{i,j}(t)) + c_2 r_2 (gb_{i,j}(t) - x_{i,j}(t)) \quad (1)$$

$$x_{i,j}(t+1) = x_{i,j}(t) + v_{i,j}(t+1) \quad (2)$$

其中, $Pb_i(t) = (pb_{i,1}(t), pb_{i,2}(t), \dots, pb_{i,D}(t))$ 和 $Gb_i(t) = (gb_{i,1}(t), gb_{i,2}(t), \dots, gb_{i,D}(t))$ 分别代表第 i 个粒子的个人最佳和全局最佳的位置。另外, c_1 和 c_2 是两个学习因子,是惯性权值,这两个随机数 r_1 和 r_2 在区间 $[0, 1]$ 中生成。其中, c_1 、 c_2 和 ω 是控制种群不同进化状态的关键参数[8],即全球勘探和局部开发。例如,如果 c_1 和 ω 较大,而 c_2 较小,则种群倾向于扩大搜索范围,以改善全球探索。相反,当 c_1 和 ω 较小时, c_2 较大时,该算法将具有更好的局部搜索能力。这样,就可以通过调整飞行参数来实现相应的进化能力。

3. 基于全局边际排序和自适应参数的改进算法(GGRAW)的介绍

3.1. 外部存档集的维护

多目标优化算法中提取非劣势个体的方法主要包括基于 Pareto 的优势策略[9],大部分算法都使用的是非 Pareto 的方法。然而,对于非 Pareto 的方法有些常见的缺点,比如极端点的排序值可能太靠近边缘,但是这些极端点的值对于后期的更新有着很重要的作用,还有在排序的过程中被选择的非支配解重复值太多,然而全局一般排序更好的解决了这一点。下面给出全局边际排序的定义。

定义 1. 全局边际排序

将个体的全局边际排序定义为所有目标值的差值之和,如等式中所述(3)

$$\text{GMR}(x_i) = \sum_{x_j \neq x_i} \max\left(\left(\prod_{m=1}^M f_m(x_i) - \prod_{m=1}^M f_m(x_j)\right), 0\right) \quad (3)$$

其中, x_i 和 x_j 是两种不同的解决方案, M 是目标的数量。随着帕累托主导地位的概念, $\text{GMR}(x_i)$ 越小, 越多的解决方案将占主导地位。根据等式(3), 对于空间中的任何两个个体, 当且仅当 $\text{GMR}(x_i) < \text{GMR}(x_j)$ 时, x_i 都优于 x_j 。如果 $\text{GMR}(x_i) = 0$, x_i 将不会在种群空间中被任何其他个体所主导。下面通过一个实例来说明它的优势性

定义 2. 全局密度(密度估计器)

$$\text{GD}(x_i) = \sum_{j=1}^{\text{pop}} d_{i,j} \quad (i \neq j) \quad (4)$$

这里, $\text{GD}(x_i)$ 表示粒子 x_i 的整体密度, $d_{i,j}$ 是 x_i 和 x_j 之间的欧氏距离。 $\text{GD}(x_i)$ 值越大, x_i 周围的粒子越少, 个体间差异越大, 分布越好。

定义 3. 全局一般排序

$$\text{GGR}(x_i) \triangleq \frac{\text{GMR}(x_i)}{\text{GD}(x_i)} \quad (5)$$

这里的 $\text{GGR}(x_i)$ 表示 x_i 的全局一般排名。 $\text{GGR}(x_i)$ 越小, x_i 的优势就越大, 这也表明 x_i 的 GD 分布越好。根据等式(5)获得的全局一般排名, 我们采取一种策略, 将前半放入外部档案, 消除后半部分, 以获得人口的最优前沿。这个过程一直持续到算法终止。

3.2. 自适应参数

在 MOPSO 中, 飞行参数、惯性权重 ω 、加速度系数 c_1 和 c_2 对于实现种群开发和勘探之间的平衡非常重要。而全局搜索和局部细化与收敛性和多样性性能直接相关。现有的模型采用线性或非线性迭代递减策略动态调整飞行参数[10]。然而, 为了正确管理算法的性能, 根据进化信息调整这些参数是非常必要的。在 GGRW 中, 根据[11], 设计了 ω 、 c_1 和 c_2 的独立自适应调整策略。为了自适应地调整飞行参数, 需要对种群进行综合评估, 以获得其进化信息。本文计算的进化信息包括种群进化能力, 每个粒子的进化能力和进化效率。

$$U_i^{gb}(t) = \sqrt{\sum_{j=1}^m (f_{i,j}^{gb}(t) - f_{i,j}^{gb}(t-1))^2}$$

$$U_i(t) = \sqrt{\sum_{j=1}^m (f_{w,j}(t) - f_{i,j}(t))^2} / \sqrt{\sum_{j=1}^m (f_{w,j}(t) - f_{b,j}(t))^2}$$

$$\varphi_i(t+1) = \sqrt{(U_i^{gb}(t))^2 + (U_i(t))^2}$$

这里 $U^{gb}(t) = (U_1^{gb}(t), U_2^{gb}(t), \dots, U_n^{gb}(t))$ 表示种群的进化能力, $U_i(t)$ 表示第 i 个粒子的进化能力。进化能力描述了种群或单个粒子找到比当前粒子更好的解决方案的能力。 $\varphi_i(t+1)$ 是进化效率, 它是结合种群进化来评价粒子进化能力的综合指标。对于所涉及的目标值, $f_{i,j}(t)$ 表示第 i 个粒子的第 j 个目标值, $f_{i,j}^{gb}(t)$ 表示全局领先者的第 j 个目标值。而 $f_{w,j}(t)$, $f_{b,j}(t)$ 分别是总体中第二差解和第二优解的第 j 个目标值。需要注意的是, 所涉及的客观值都在 $[0, 1]$ 的范围内标准化了。

根据上述计算的演化信息, 可以通过(ε 为常数)得到第 i 个粒子的自适应参数

$$R_i(t+1) = 1 / \sqrt{(\varphi_i(t+1))^2 + \varepsilon} \quad (6)$$

在优化过程的早期阶段, 全局搜索对种群进化很重要, 而也需要开发在后期阶段, 细化解决方案[5]。

相应的, ω 和 c_1 越大, c_2 越小的条件有利于全局勘探, 而 ω 和 c_1 越小, c_2 越大的条件促进局部开发。因此, 我们设计了独立的自适应调整策略来调整每个粒子的 ω 、 c_1 和 c_2 。

$$\omega_i(t+1) = \omega_{\max} - (\omega_{\max} - \omega_{\min}) \times \sin(R_i(t+1)\pi/2) \times \frac{t}{T} \quad (7)$$

$$c_{1i}(t+1) = c_{1\max} - c_{1\max} \times \sin(R_i(t+1)\pi/2) \times \frac{t}{T} \quad (8)$$

$$c_{2i}(t+1) = c_{2\min} + c_{2\min} \times \sin(R_i(t+1)\pi/2) \times \frac{t}{T} \quad (9)$$

其中 ω_{\max} , ω_{\min} 为 ω 的最大值和最小值, $c_{1\max}$ 为 c_1 的最大值, $c_{2\min}$ 为 c_2 的最小值, T 为预先设定的最大总体迭代次数。这样, ω 和 c_1 的值就会很大, 而 c_2 的值在早期进化中就会很小。因此, 将促进全局勘探, 以扩大搜索范围。而在后期的进化过程中, ω 和 c_1 会很小, 而 c_2 会很大。相应地, 可以促进局部开发的种群收敛。这种参数调整不仅推动了近似 PF 向前发展, 而且确保了更多样化的解。

3.3. 个体最优解和全局最优解的选取

在多目标粒子群优化算法中个体最优解的选取(Pbest)和全局最优解(Gbest)的选取是决定算法是否有效的关键性因素, 因此对于选取个体最优解和全局最优解至关重要。对于个体最优解的选取通过由公式(3)计算出来的值与当前由公式(3)计算出来的值进行比较, 如果当前粒子的值小于前一代粒子的值, 则选取当前粒子的值作为 Pbest, 否则把前一代的粒子的值作为 Pbest。然而对于全局最优解用外部存档集即可, 算法的框架为图 1。

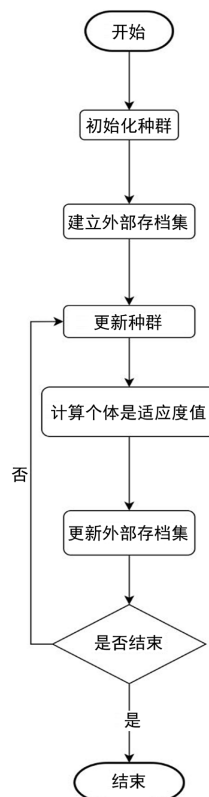


Figure 1. Flow chart of GGRAW algorithm

图 1. GGRAW 算法流程图

具体步骤如下:

- 1) 初始化种群 P 且 $P_{best} = P$
- 2) 通过全局一般排序建立外部存档集
- 3) 随机生成 $G_{best}(N)$
- 4) 随机生成自适应参数 ω 、 c_1 和 c_2 个数为 N
- 5) 设置终止条件
- 6) 更新种群 P
- 7) 通过全局一般排序更新外部存档集
- 8) 更新 P_{best} , G_{best}
- 9) **for** $i=1:N$
- 10) 通过种群 P 和 G_{best} 更新 R_i
- 11) 更新参数 ω 、 c_1 和 c_2
- 12) 结束 **for**
- 13) 判断是否满足终止条件, 若满足则迭代结束, 否则返回第 5 步
- 14) 输出外部存档集

4. 实验结果与分析

4.1. 评价指标

在本文中, 我们采用了两个综合的指标, 超体积(HV)和迭代距(IGD)来量化测试算法。虽然这两个指标都能表达关于收敛性和多样性的综合信息, 但一个指标不能充分评价实验算法。IGD 指标主要通过计算测试算法得到的近似 PF 上沿真帕累托前沿的均匀分布解到其最近解的平均距离来评估算法[11]。IGD 的公式如(18), 其中 Ω 表示在真 PF 上均匀采样的一组点, Θ 是通过测试算法得到的一组最优解。此外, $|\Omega|$ 表示采样点的数量, $dis(x, \Theta)$ 是 Θ 中 x 与最近点之间的最小欧氏距离。在计算 IGD 时, 参考采样点是由 Das 和 Dennis 在[12]中提出的系统方法生成的。高压指示器计算所有超立方体的并集体积, 以评估算法[13]。由所得到的解集成的解和一个参考点 a 形成了超立方体。HV 的计算公式如(19), 其中, δ 表示勒贝格测度, 用于计算超立方体的体积。 θ_i 为第 i 个解及对应参考点形成的超立方体。在计算 HV 时, 参考[14]中的建议, 首先使用理想点和最低点对得到的非支配解的目标值进行归一化。 z_{ideal} 和 z_{nadir} 分别是真 PF 的理想点和最低点。然后将参考点指定为(1.1、1.1、...、1.1)

$$IGD(\Theta, \Omega) = \frac{\sum_{x \in \Omega} dis(x, \Theta)}{|\Omega|} \quad (18)$$

$$HV = \delta \left(\bigcup_{i=1}^{|\Omega|} \theta_i \right) \quad (19)$$

4.2. 实验条件及参数设置

为了更加有效的验证 GGRaw 算法的性能, 本文选取了 3 个二维的测试函数(ZDT1, ZDT2, ZDT3) [15] 和两个三维的测试函数(DTLZ2 [16], WFG4 [17])对这 4 种算法(MOPSO, dMOPSO, SMP SO)进行了对比测试。关于 GGRaw 的算法设置参数如下: 种群规模 100, 外部存档规模为 100, 迭代次数为 1000 代, $\omega_{max} = 0.5$, $\omega_{min} = 0.42$, $c_{1max} = 2.33$, $c_{2min} = 1.21$, 其他的算法参数与它被提出时的参数保持一致。

4.3. 算法的仿真分析结果

图 2~6 分别展示了 dMOPSO, MOPSO, SMP SO, GGRaw 四种算法在 4.2 中所提到的测试函数上的仿

真结果图。从图 2~6 中可以看出对于算法 GGRAW 所获得的非劣解的收敛性除了 SMPSO 在测试函数 ZDT2 的收敛性与 GGRAW 算法的收敛性相差不大外，其余的情况明显好于其余三算法。从图 5，图 6 可以很好的看出，GGRAW 算法在分布性上也明显好于其余三个算法。

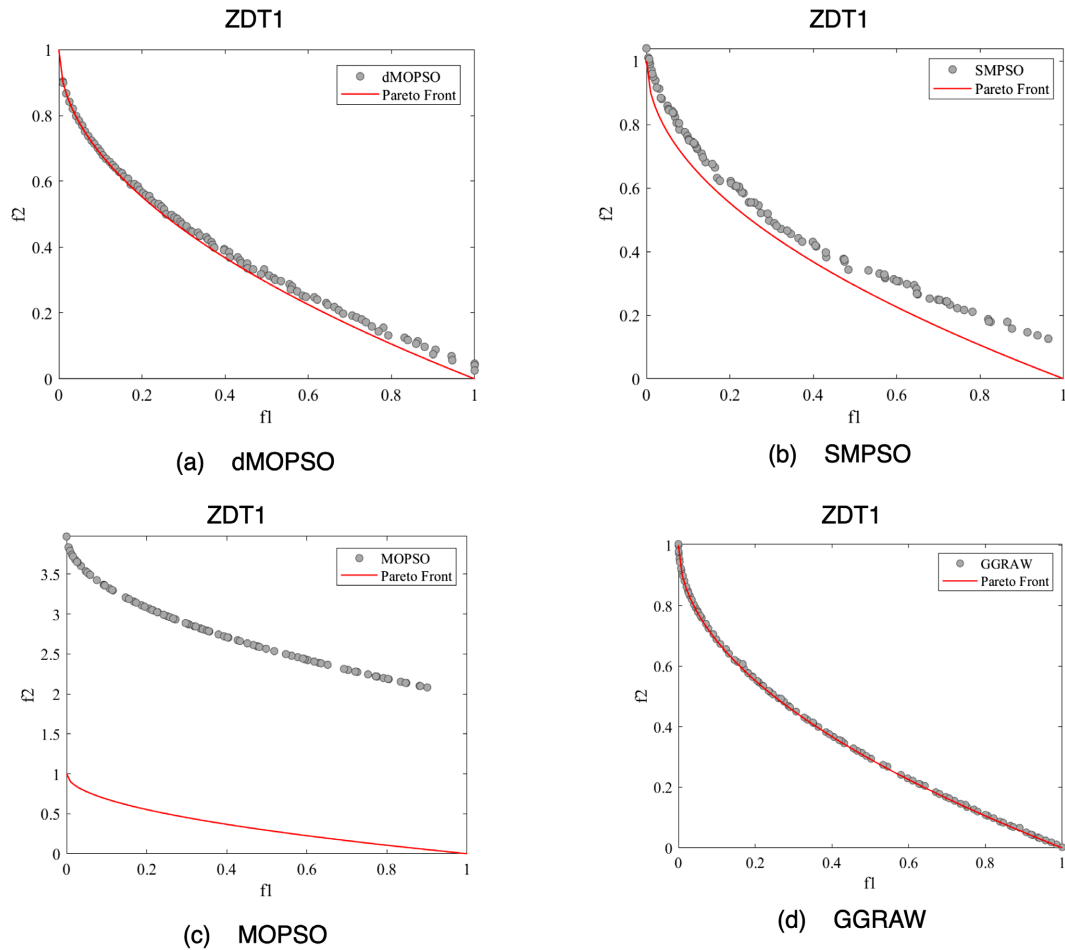
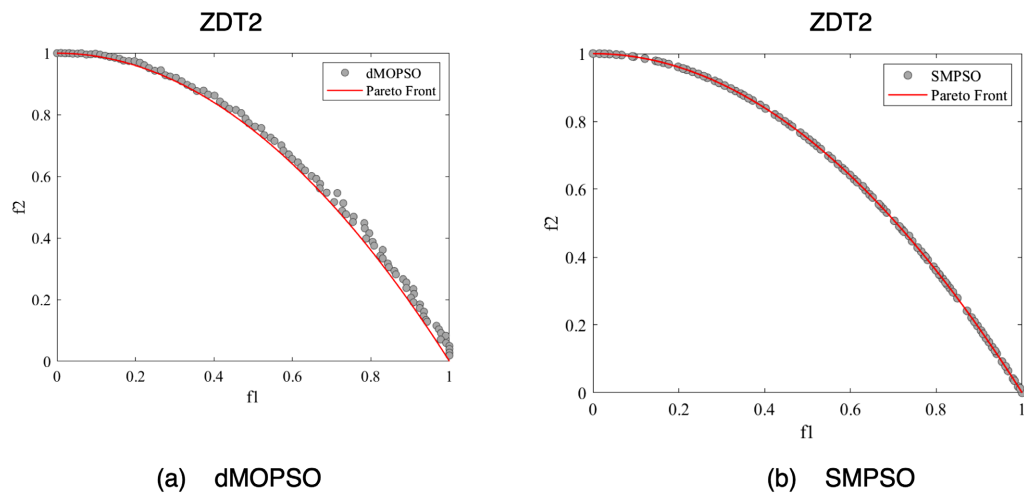


Figure 2. Simulation results of algorithm on test function ZDT1

图 2. 算法在测试函数 ZDT1 上的仿真结果



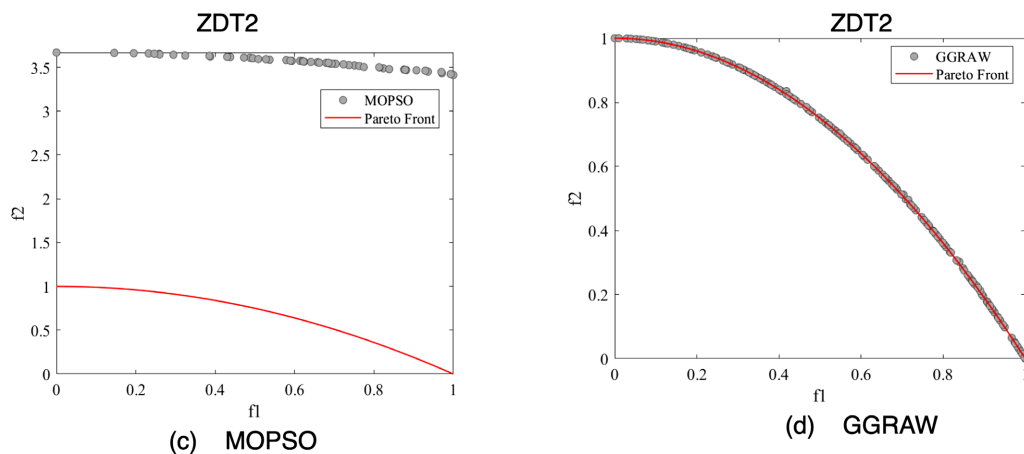


Figure 3. Simulation results of algorithm on test function ZDT2
 图 3. 算法在测试函数 ZDT2 上的仿真结果

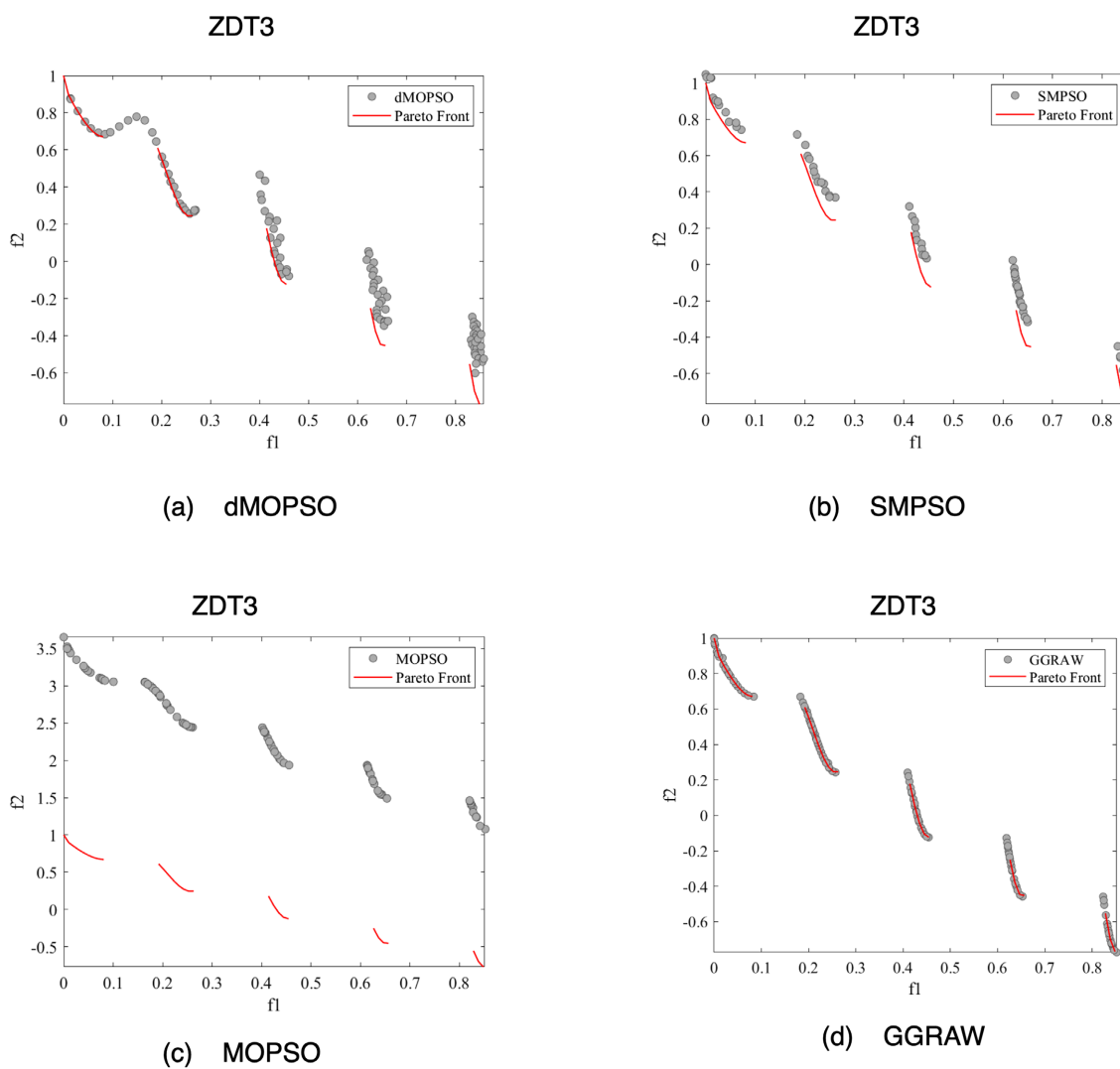


Figure 4. Simulation results of algorithm on test function ZDT3
 图 4. 算法在测试函数 ZDT3 上的仿真结果

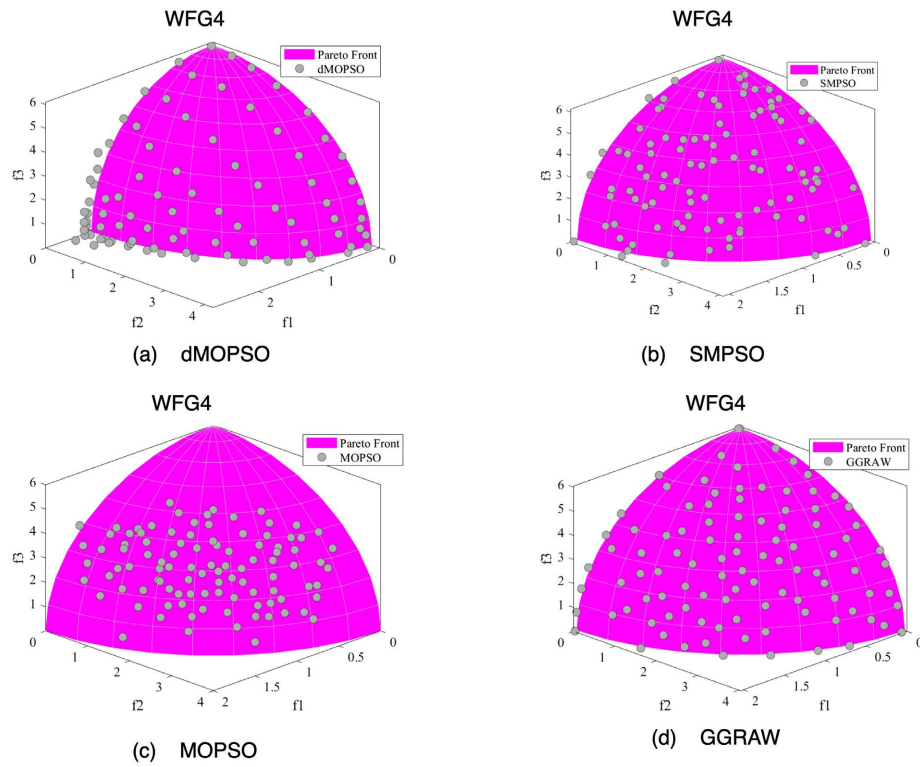


Figure 5. Simulation results of algorithm on test function WFG4
 图 5. 算法在测试函数 WFG4 上的仿真结果

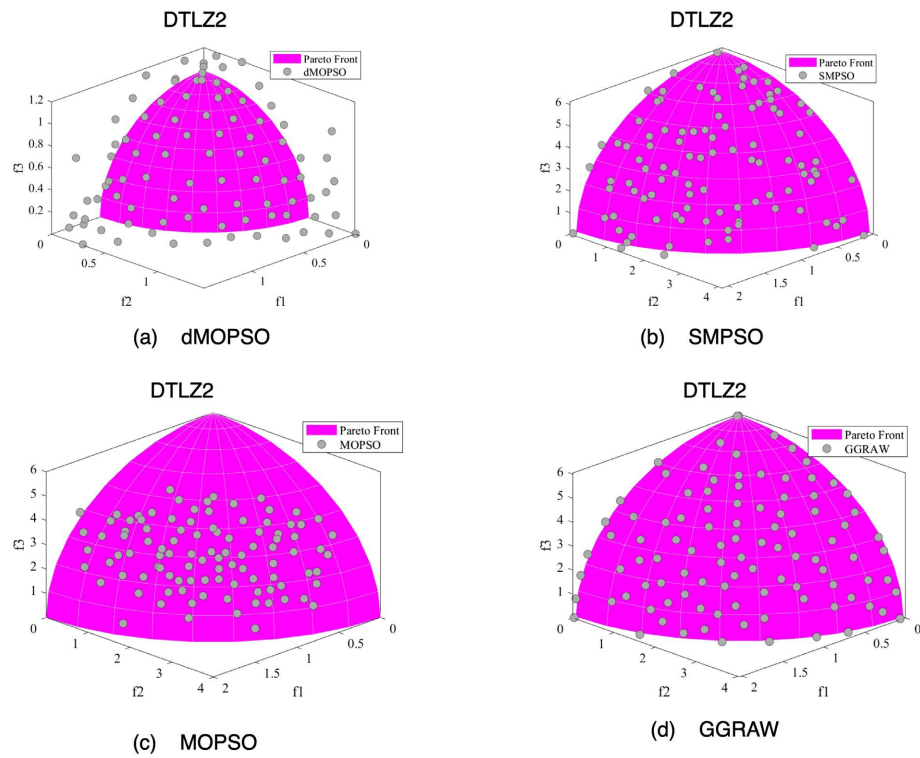


Figure 6. Simulation results of algorithm on test function DTLZ2
 图 6. 算法在测试函数 DTLZ2 上的仿真结果

4.4. 算法的收敛性与多样性的综合分析

表 1 和表 2 分别展示了 dMOPSO, MOPSO, SMPSO 和 GGRW 四种算法在测试函数 ZDT, DTLZ, WFG 上独立运行 50 次所获得的综合指标 IGD 和 HV 的标准差和平均值。需要注意的是,表中的符号“+”表示其他 MOPSOs 的结果明显优于 GGRW 的结果,而“-”表示结果明显差于 GGRW。而“=”表示比较算法的结果,结果与 GGRW 相似。下面将根据者四种算法所获得的 IGD 和 HV 指标,对四种算法的收敛性和多样性进行综合分析。

Table 1. Indexes of various algorithms on IGD

表 1. 各种算法在 IGD 上的指标

函数	IGD 指标	dMOPSO	SMPSO	MOPSO	GGRW
ZDT1	Mean (Std)	3.0131e-2 (8.57e-3) -	5.5569e-2 (6.22e-2) -	1.6479e+0 (1.02e-1) -	5.1260e-3 (4.18e-4)
ZDT2	Mean (Std)	6.6052e-2 (1.37e-1) -	2.6439e-2 (6.32e-2) -	3.2009e+0 (1.87e-1) -	5.1260e-3 (4.18e-5)
ZDT3	Mean (Std)	2.8915e-2 (6.79e-3) -	1.6664e-1 (9.45e-2) -	1.2587e+0 (1.18e-1) -	5.1260e-3 (4.18e-6)
DTLZ2	Mean (Std)	1.4467e-1 (9.29e-3) -	9.5469e-2 (7.27e-3) -	1.5866e-1 (3.19e-2) -	5.1260e-3 (4.18e-7)
WFG4	Mean (Std)	4.0273e-1 (2.85e-2) -	3.3999e-1 (1.60e-2) -	6.4192e-1 (1.59e-1) -	5.1260e-3 (4.18e-8)
+/-/=		0/5/0	0/5/0	0/5/0	

Table 2. Indexes of various algorithms on HV

表 2. 各种算法在 HV 上的指标

函数	HV 指标	dMOPSO	SMPSO	MOPSO	GGRW
ZDT1	Mean (Std)	6.8507e-1 (1.02e-2) -	6.5054e-1 (7.88e-2) -	0.0000e+0 (0.00e+0) -	7.1717e-1 (7.56e-4)
ZDT2	Mean (Std)	3.7806e-1 (8.02e-2) -	4.1649e-1 (6.99e-2) =	0.0000e+0 (0.00e+0) -	4.4254e-1 (9.29e-4)
ZDT3	Mean (Std)	6.0498e-1 (9.71e-3) +	5.5176e-1 (6.34e-2) -	2.8529e-4 (1.11e-3) -	5.9886e-1 (8.00e-4)
DTLZ2	Mean (Std)	3.6739e-1 (1.58e-2) -	4.4960e-1 (1.57e-2) -	3.9751e-1 (3.52e-2) -	5.4104e-1 (3.10e-3)
WFG4	Mean (Std)	4.2306e-1 (9.51e-3) -	4.4529e-1 (1.03e-2) -	3.5712e-1 (3.28e-2) -	4.8397e-1 (4.06e-3)
+/-/=		1/4/0	0/4/1	0/5/0	

通过表 1 和表 2 不难看出, GGRW 算法所获得的 IGD 和 HV 指标整体上是明显优于算 dMOPSO, SMPSO, MOPSO, 说明了 GGRW 算法有较好的综合性能, 从表 1 看可以看出 GGRW 算法在表中的测试函数所获得 IGD 的值都是优于其余三个算法的, 从表 2 中来看, 在测试函数 ZDT2 中算法 SMPSO

与 GGRW 算法在 HV 相差不大, 在 ZDT3 上对于指标 HV 来说算法 dMOPSO 是优于 GGRW, 但是从总体来看 GGRW 算法综合性能是好于与之相比的 3 种算法。

5. 结论

为了更加有效地提高种群的多样性和收敛性, 本文提出了基于全局边际排序和自适应参数调节的多目标粒子群算法(GGRW)。该算法通过全局边际排序的优势性对外部存档的规模进行了有效的维护, 并且在自适应参数的加持下更加使得在种群中的优秀的粒子在随后的更新迭代中得以保存和发展, 有效提升了种群的多样性, 同时也提高了粒子收敛到真实的 Pareto 的前沿的概率。将 GGRW 算法在 dMOPSO, MMOPSO 和 SMPSO 三种算法在同一个环境下进行了仿真实验, 实验的结果表明: GGRW 算法与其余 3 种算法相比较, 在收敛性和综合性能上明显的改善, 这就意味着 GGRW 算法可以作为求解多目标优化问题的有效算法。

参考文献

- [1] Chen, X., Du, W. and Qian, F. (2014) Multi-Objective Differential Evolution with Ranking-Based Mutation Operator and Its Application in Chemical Process Optimization. *Chemometrics and Intelligent Laboratory Systems*, **136**, 85-96. <https://doi.org/10.1016/j.chemolab.2014.05.007>
- [2] Han, D., Du, W., Du, W., Jin, Y. and Wu, C. (2019) An Adaptive Decomposition Based Evolutionary Algorithm for Many-Objective Optimization. *Information Sciences*, **491**, 204-222. <https://doi.org/10.1016/j.ins.2019.03.062>
- [3] Sun, Y.N., Xue, B., Zhang, M.J. and Yen, G.G. (2019) A New Two-Stage Evolutionary Algorithm for Many-Objective Optimization. *IEEE Transactions on Evolutionary Computation*, **23**, 748-761. <https://doi.org/10.1109/TEVC.2018.2882166>
- [4] Palakonda, V., Mallipeddi, R. and Suganthan, P.N. (2021) An Ensemble Approach with External Archive for Multi- and Many-Objective Optimization with Adaptive Mating Mechanism and Two-Level Environmental Selection. *Information Sciences*, **555**, 164-197. <https://doi.org/10.1016/j.ins.2020.11.040>
- [5] Wang, L., Pan, X., Shen, X., Zhao, P. and Qiu, Q. (2021) Balancing Convergence and Diversity in Resource Allocation Strategy for Decomposition-Based Multi-Objective Evolutionary Algorithm. *Applied Soft Computing*, **100**, Article ID: 106968. <https://doi.org/10.1016/j.asoc.2020.106968>
- [6] Kennedy, J. and Eberhart, R. (1995) Particle Swarm Optimization. *Proceedings of ICNN'95-International Conference on Neural Networks*, **4**, 1942-1948. <https://doi.org/10.1109/ICNN.1995.488968>
- [7] Cheng, S.X., Zhan, H., Yao, H.Q., Fan, H.Y. and Liu, Y. (2021) Large-Scale Many-Objective Particle Swarm Optimizer with Fast Convergence Based on Alpha-Stable Mutation and Logistic Function. *Applied Soft Computing*, **99**, Article ID: 106947. <https://doi.org/10.1016/j.asoc.2020.106947>
- [8] Han, H.G., Lu, W., Zhang, L. and Qiao, J.F. (2018) Adaptive Gradient Multiobjective Particle Swarm Optimization. *IEEE Transactions on Cybernetics*, **48**, 3067-3079. <https://doi.org/10.1109/TCYB.2017.2756874>
- [9] Biswas, S., Das, S., Suganthan, P.N. and Coello, C.A.C. (2014) Evolutionary Multiobjective Optimization in Dynamic Environments: A Set of Novel Benchmark Functions. 2014 *IEEE Congress on Evolutionary Computation (CEC)*, Beijing, 6-11 July 2014, 3192-3199. <https://doi.org/10.1109/CEC.2014.6900487>
- [10] Zhang, X.-H., Meng, H.-Y. and Jiao, L.-C. (2005) Intelligent Particle Swarm Optimization in Multiobjective Optimization. 2005 *IEEE Congress on Evolutionary Computation*, **1**, 714-719. <https://doi.org/10.1109/CEC.2005.1554753>
- [11] Cui, Y.Y., Qiao, J.F. and Meng, X. (2020) Multi-Stage Multi-Objective Particle Swarm Optimization Algorithm Based on the Evolutionary Information of Population. 2020 *Chinese Automation Congress (CAC)*, Shanghai, 6-8 November 2020, 3412-3417. <https://doi.org/10.1109/CAC51589.2020.9326666>
- [12] Das, I. and Dennis, J.E. (1998) Normal-Boundary Intersection: A New Method for Generating the Pareto Surface in Nonlinear Multicriteria Optimization Problems. *SIAM Journal on Optimization*, **8**, 631-657. <https://doi.org/10.1137/S1052623496307510>
- [13] Chen, N., Chen, W.N., Gong, Y.J., Zhan, Z.H., Zhang, J., Li, Y. and Tan, Y.S. (2015) An Evolutionary Algorithm with Double-Level Archives for Multiobjective Optimization. *IEEE Transactions on Cybernetics*, **45**, 1851-1863. <https://doi.org/10.1109/TCYB.2014.2360923>
- [14] Ishibuchi, H., Setoguchi, Y., Masuda, H. and Nojima, Y. (2017) Performance of Decomposition-Based Many-Objective Algorithms Strongly Depends on Pareto Front Shapes. *IEEE Transactions on Evolutionary Computation*, **21**, 169-190.

- <https://doi.org/10.1109/TEVC.2016.2587749>
- [15] Ler, E., Deb, K. and Thiele, L. (2000) Comparison of Multiobjective Evolutionary Algorithms: Empirical Results. *Evolutionary Computation*, **8**, 173-195. <https://doi.org/10.1162/106365600568202>
- [16] Deb, K., Thiele, L., Laumanns, M. and Zitzler, E. (2002) Scalable Multi-Objective Optimization Test Problems. *Proceedings of the 2002 Congress on Evolutionary Computation*, **1**, 825-830. <https://doi.org/10.1109/CEC.2002.1007032>
- [17] Huband, S., Hingston, P., Barone, L. and While, L. (2006) A Review of Multiobjective Test Problems and a Scalable Test Problem Toolkit. *IEEE Transactions on Evolutionary Computation*, **10**, 477-506. <https://doi.org/10.1109/TEVC.2005.861417>