黄水木, 刘能现. 基于混合竞争机制的多目标正余弦算法[J]. 智能计算机与应用,2026,16(5):140-146. DOI:10.20169/j.issn.2095-2163.24082201

基于混合竞争机制的多目标正余弦算法

黄水木[1], 刘能现[1,2]
(1 四创科技有限公司, 福州 350100; 2 福州大学 研究生院, 福州 350108)

摘 要: 针对部分多目标算法难以同时获得收敛性和多样性较优的非支配解问题,本文提出一种基于混合竞争机制的多目标正余弦算法(A multi-objective sine-cosine algorithm based on a hybrid competition mechanism, HCMOSCA)。该算法在进化的前半阶段采用基于距离的竞争机制,侧重于种群的收敛性;进化的后半阶段采用基于角度的竞争机制,侧重于种群的多样性。本文在 DTLZ 和 ZDT 两个测试集上验证了 HCMOSCA 的性能。实验结果表明,HCMOSCA 比 4 个对比算法有更优的性能表现,能够获得收敛性和多样性俱佳的非支配解。

关键词: 混合竞争机制; 多目标优化; 正余弦算法; 收敛性; 多样性
中图分类号: TP301 **文献标志码**: A **文章编号**: 2095-2163(2026)05-0140-07

A multi-objective sine-cosine algorithm based on a hybrid competition mechanism

HUANG Shuimu[1], LIU Nengxian[1,2]
(1 ISTRONG Technology Co., Ltd., Fuzhou 350100, China;
2 Graduate School, Fuzhou University, Fuzhou 350108, China)

Abstract: In view of the difficulty that some multi-objective algorithms have in obtaining non-dominated solutions with good convergence and diversity, this paper proposes a multi-objective sine-cosine algorithm based on a hybrid competition mechanism (HCMOSCA). HCMOSCA adopts a distance-based competition mechanism in the early stage of evolution, focusing on the convergence of the population; and adopts an angle-based competition mechanism in the second half of the evolution, focusing on the diversity of the population. The performance of HCMOSCA is verified on the DTLZ and ZDT test sets. The experimental results show that HCMOSCA has better performance than the four compared algorithms and can obtain non-dominated solutions with good convergence and diversity.

Key words: hybrid competition mechanism; multi-objective optimization; sine-cosine algorithm; convergence; diversity

0 引 言

近年来,元启发式算法在解决实际问题方面的优势引起了研究人员的关注。由于计算成本低、鲁棒性强,这些算法被广泛应用于各个领域的优化问题中[1-4]。其中,多目标优化问题(Multi-objective Optimization Problem,MOP)是一类极具挑战性的实际优化问题。如果一个问题存在多个需要优化的目标,则称该问题为多目标优化问题。一般来说,多目标优化问题可表示如下:

$$\begin{aligned} &\min\ F(x)=(f_1(x),f_2(x),\cdots,f_M(x))\\ &\text{s.t. } g_i(x)\leqslant 0\ (i=1,2,\cdots,J)\\ &\quad h_i(x)=0\ (i=1,2,\cdots,K)\\ &\quad x=(x_1,x_2,\cdots,x_N)\\ &\quad L_i\leqslant x_i\leqslant U_i,\ i=1,2,\cdots,D \end{aligned} \tag{1}$$

其中,$f_i(x)$ 为目标函数; M 为目标函数个数; $g_i(x)$、$h_i(x)$ 分别为不等式约束与等式约束函数; J、K 分别为其约束个数; L_i、U_i 分别为第 i 个变量的下界、上界。

假定两个可行解 a 和 b, 若满足 $\forall_i \in \{1,2,\cdots,M\}$、$f_i(a)\leqslant f_i(b)$ 且 $\exists_j \in \{1,2,\cdots,M\}$、

作者简介: 黄水木(1984—),男,学士,工程师。主要研究方向:软件设计,智能计算及其应用。
通信作者: 刘能现(1983—),男,博士,高级工程师。主要研究方向:智能计算及其应用。Email:187064463@qq.com。

收稿日期: 2024-08-22

$f_j(a) < f_j(b)$，则称 a 支配 b。如果不存在其他解支配解 a，那么称解 a 为帕累托最优解。决策空间中全部的帕累托最优解组成帕累托最优解集（Pareto-optimal Set，PS），这些解的目标值所形成的边界称为帕累托前沿（Pareto-optimal Front，PF）。

在过去的二十多年里，多目标进化算法（Multi-Objective Evolutionary Algorithms，MOEAs）因其在处理多目标优化问题（MOP）时所展现的卓越性能，受到了广泛的关注。目前，学者们已经提出了众多基于不同元启发式算法的多目标进化算法，如多目标遗传算法[5-6]、多目标粒子群算法[7-8]、多目标平衡优化黏菌算法[9]、多目标差分进化算法等[10-11]，这些算法旨在提高求解多目标优化问题的效率和效果。尽管如此，受限于其设计原理，这些多目标进化算法仍无法满足当前的复杂优化需求。为此，本文提出一种新的混合竞争机制的多目标正余弦优化算法（HCMOSCA）。该算法基于帕累托支配的多目标算法框架，并混合采用角度竞争机制和距离竞争机制来设计正余弦优化算法（Sine-Cosine Algorithm，SCA）的位置更新公式，从而提升算法在多目标优化问题中的性能。

1　正余弦优化算法（SCA）

正余弦优化算法（Sine Cosine Algorithm，SCA）[12-14]是由 Seyedali Mirjalili 于 2016 年提出的一种基于种群的元启发式优化技术，其灵感来自正弦和余弦函数的数学特性。在求解优化问题的过程中，SCA 利用正弦和余弦函数的周期性和规律性来指导搜索代理（或称为个体、候选解）在搜索空间中的位置更新，从而逐渐获得问题的最优解。SCA 中代理的位置更新公式如下：

$$\begin{cases} x_i^{t+1} = x_i^t + r_1 \times \sin(r_2) \times \left| r_3 p_i^t - x_i^t \right|, & r_4 < 0.5 \\ x_i^{t+1} = x_i^t + r_1 \times \cos(r_2) \times \left| r_3 p_i^t - x_i^t \right|, & r_4 \geqslant 0.5 \end{cases} \tag{2}$$

其中，x_i^{t+1}、x_i^t 分别为第 $(t+1)$ 次和第 t 次迭代的第 i $(i = 1,2,\cdots,N)$ 个代理；p_i^t 表示第 t 次迭代的最佳代理（目标点）；r_1、r_2、r_3 和 r_4 为随机控制参数，这些参数的集成旨在确保算法在搜索过程中能够避免陷入局部最优解，并平衡搜索过程的探索和开发。

参数 r_1 在迭代搜索过程中承担平衡探索与开发的作用：在迭代搜索过程的前半阶段有助于算法的探索，意味着算法能够广泛地在解空间中搜索，在迭代搜索过程的后半阶段有助于算法的开发，强调的是在已探索的区域内进行更精细的搜索。为了平衡探索和开发，r_1 从预设常数 a 自适应地减小到 0，公式如下：

$$r_1 = a - t\frac{a}{T} \tag{3}$$

其中，a 为常数，t、T 分别表示当前迭代次数和最大迭代次数。

r_2 为方向参数，指示当前代理是朝向最佳代理移动还是向最佳代理外移动，r_2 的取值范围为 $[0, 2\pi]$。r_3 为权重参数，用于随机加强（$r_3 > 1$）或削弱（$r_3 < 1$）最佳代理对其他候选代理的影响，r_3 的取值范围为 $[0,2]$。r_4 是一个均匀随机参数，其取值范围为 $[0,1]$，在公式（2）中用于切换正弦和余弦函数。

SCA 的求解过程从初始化步骤开始，SCA 以随机方式生成代理群体（初始解）。这些代理通过公式（2）迭代搜索更新，当满足终止条件时，迭代过程停止并输出最优解。

2　基于混合竞争机制的多目标正余弦优化算法（HCMOSCA）

2.1　算法框架

HCMOSCA 算法结合了正余弦算法（SCA）的搜索机制与经典多目标优化算法 SPEA[15]的环境选择策略，是一种面向多目标优化问题的元启发式算法。该算法采用了一个简洁而有效的框架，通过迭代进化过程搜索问题的非支配解集。算法框架如算法 1 所示，其中主循环涉及两个模块，即基于混合竞争机制的 SCA 位置更新和多目标环境选择。HCMOSCA 的主要框架如图 1 所示。

算法 1　HCMOSCA 框架

输入　种群大小 N，决策变量维数 D，最大函数评估次数 MaxFES，当前函数评估次数 FES。

1：随机初始化代理种群 P；

2：While（FES<MaxFES）do

3：$P' \leftarrow$ 基于混合竞争机制 SCA 位置更新 (P)

4：$P \leftarrow$ 多目标环境选择 (P, P')

5：End while

输出　种群 P 中所有非支配代理

2.2　基于混合竞争机制的 SCA 位置更新

基于混合竞争机制的 SCA 位置更新主要包括创建精英代理集合、基于距离或角度竞争机制的代理比较、代理位置更新。算法 2 给出了基于混合竞争机制的 SCA 位置更新的伪代码。

算法 2　基于混合竞争机制的 SCA 位置更新

输入 当前代理种群 P, 精英代理集合大小 α

$P' \leftarrow \Phi$;

1:精英代理集合 $E \leftarrow$ 根据前沿面序号和拥挤距离从 P 中选择 α 个精英代理;

2:For P 中的每个代理 x_i do

3:$\{a,b\} \leftarrow$ 从 E 中随机选择两个个体;

4:If (FES<MaxFES/2) do

5:采用距离竞争机制计算 a,b 与原点距离;

6:Else

7:采用角度竞争机制计算 a,b 与 x_i 角度大小;

8:End if

9:If a 的距离(角度) < b 的距离(角度)do

10: $x_w = a$; //竞争胜利者

11: $x_l = b$; //竞争失败者

12: Else

13: $x_l = a$;

14: $x_w = b$;

15: End if

16: 随机更新 r_1、r_2 和 r_4;

17: 用公式(4)生成新的 x_i';

18: $P' \leftarrow P' \cup \{x_i'\}$;

19: End for

输出 后代种群 P'

首先,构建一个精英代理集合 E, 该集合在所提出的 SCA 位置更新中扮演着重要角色,其提供了成对竞争中采用的候选代理,用来引导群体搜索。与 CMOPSO 类似[7],精英代理集采用 NSGA-II[5] 中提出的非支配排序和基于拥挤距离的排序方法来选择精英代理。该方法旨在确保代理集合中代理的多样性和收敛性得到良好的平衡,从而在优化过程中保持更高的效率和效果。

接着,从精英代理集合 E 中随机选择两个代理 a 和 b。在种群进化的前半阶段,算法采用基于欧拉距离竞争机制,分别计算代理 a 和 b 与原点的距离,距离小的作为胜利代理。在种群进化的后半阶段,算法采用基于角度竞争机制,分别计算代理 a 和 b 与当前代理 x_i 的夹角,夹角小的作为胜利代理。胜利代理记为 x_w ,胜利代理 x_w 参与当前代理的位置更新,改进的位置更新公式如下:

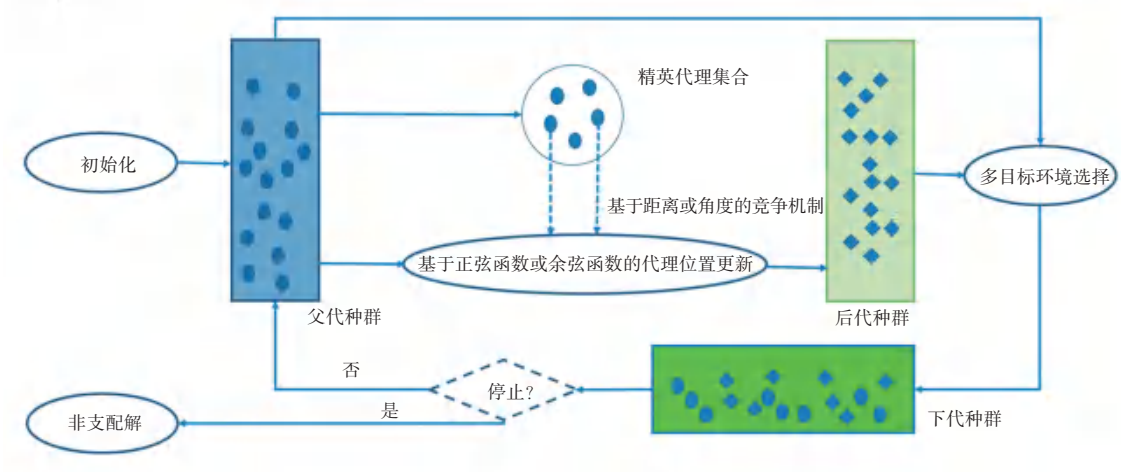


图 1 HCMOSCA 算法框架

Fig. 1 The framework of HCMOSCA algorithm

$$\begin{cases} x_i' = x_i + r_1 \times \sin(r_2) \times | r_3 x_w - x_i |, & r_4 < 0.5 \\ x_i' = x_i + r_1 \times \cos(r_2) \times | r_3 x_w - x_i |, & r_4 \geqslant 0.5 \end{cases} \tag{4}$$

因多目标问题中不存在最佳代理,在提出的位置更新公式(4)中, 胜利代理 x_w 用于替换原位置更新公式(2)中的最佳代理。另一方面,精英代理集中的胜利代理是通过 NSGA-II 中的帕累托方法选择的,胜利代理更接近帕累托前沿并且密度较低,因此,所提出的公式中以胜利代理指导进化搜索的位置更新具有更好的收敛性和多样性。为保持胜利代理与前沿面的关系,参数 r_3 设为 1。

最后,将生成的新位置 x_i' 放入后代种群 P',生成 N 个新位置后,返回到算法 1,执行多目标环境选择。

3 实验设计与分析

3.1 对比算法和测试函数

为检验 HCMOSCA 的性能,将 HCMOSCA 与多个优秀多目标算法进行比较,包括:EMOSO[16]、CMOPSO[7]、NSGA-II [5]和 MOEA/D-DE[17]。

测试函数包括 DTLZ1－DTLZ7 和 ZDT1－ZDT4，DTLZ 问题[18]是可变目标函数，ZDT 问题[19]是双目标函数，DTLZ 和 ZDT 为论文作者姓的首字母组合。这些测试问题经过精心设计，考虑了多样性特征，旨在全面模拟和覆盖现实世界中可能遇到的各种复杂场景。这些问题不仅包括了不连续、凸、凹等几何特性的挑战，还涵盖了退化、多峰等优化难题，以及具有不规则帕累托前沿形状的复杂场景。因此，这些函数能够有效地评估多目标进化算法。

3.2　评价指标

本文采用两个常用的评价指标，即反转世代距离（Inverted Generational Distance，IGD）[20]和超体积（Hyper Volume，HV）[21]，评价不同对比算法的性能。IGD 值越小，说明算法性能越好；HV 值越大，说明算法性能越好。同时，为客观地对比实验结果，本文还采用了显著性水平为 0.05 的秩检验对不同对比算法进行评估。“+”“=”“－”分别表示对比算法的统计结果在显著性水平下优于、相似于和劣于本文提出的 HCMOSCA。

3.3　实验参数

实验平台采用 Tian 等[22-23]开发的 PlatEMO 平台，所有对比算法的编程语言为 MATLAB。DTLZ 问题的决策变量维数设为 $D = k + M - 1$，D 和 M 分别是决策变量和目标函数维度。对于 DTLZ1 函数的 k 值设为 5，DTLZ2－DTLZ6 函数的 k 值设为 10，对于 DTLZ7 函数的 k 值设为 20。ZDT1－ZDT3 问题的决策变量维数设为 30，ZDT4 问题的决策变量维数设为 10。对于所有 2 目标和 3 目标问题，最大函数评估次数均设置为 30 000。精英代理集合大小设为 5，种群大小设为 100。

3.4　实验结果

表 1 给出了本文提出的算法 HCMOSCA 与其他对比算法在测试函数 DTLZ 和 ZDT 上得到的 IGD 值的平均值和标准差。从表 1 可以看出 HCMOSCA 与其他算法相比有较好的性能表现，HCMOSCA 在 18 个测试函数中获得了 7 个最小的 IGD 值，EMOSO，CMOPSO，NSGA－II，MOEA/D－DE 分别获得了 2 个，1 个，5 个和 3 个最小 IGD 值。在 18 个问题中，与 EMOSO、CMOPSO、NSGA－II 和 MOEA/D－DE 相比，所提出的 HCMOSCA 分别在 13、10、13 和 9 个问题上具有更好的性能。

表 1　不同对比算法在 DTLZ 和 ZDT 测试函数上得到 IGD 的平均值与标准差

Table 1　Average and standard deviation of IGD obtained by different comparative algorithms on test functions DTLZ and ZDT

Problem	M	D	EMOSO	CMOPSO	NSGA－II	MOEA/D－DE	HCMOSCA
DTLZ1	2	6	1.882 4e+1 (3.89e+0) =	3.728 9e−1 (5.90e−1) +	**2.365 9e−3 (2.21e−4) +**	2.762 5e−3 (2.45e−3) +	1.775 0e+1 (5.73e+0)
	3	7	1.650 1e+1 (3.33e+0) −	7.573 4e+0 (4.28e+0) +	**2.757 0e−2 (1.59e−3) +**	8.062 7e−2 (1.13e−1) +	1.259 4e+1 (3.77e+0)
DTLZ2	2	11	4.392 2e−3 (7.88e−5) −	4.400 1e−3 (7.93e−5) −	5.090 4e−3 (1.77e−4) −	**3.977 8e−3 (3.09e−6) +**	4.105 4e−3 (3.17e−5)
	3	12	5.824 3e−2 (1.11e−3) −	5.788 2e−2 (1.05e−3) −	6.894 0e−2 (3.02e−3) −	7.582 2e−2 (8.12e−4) −	**5.635 2e−2 (6.56e−4)**
DTLZ3	2	11	1.827 8e+2 (1.35e+1) −	3.489 5e+1 (2.66e+1) +	**1.682 6e−2 (9.68e−3) +**	8.402 7e+0 (1.31e+1) +	1.637 0e+2 (1.93e+1)
	3	12	1.784 3e+2 (1.74e+1) −	1.023 6e+2 (3.78e+1) +	**3.879 2e−1 (6.86e−1) +**	5.086 3e+0 (9.27e+0) +	1.662 1e+2 (1.72e+1)
DTLZ4	2	11	1.765 4e−1 (3.17e−1) −	2.749 0e−1 (3.62e−1) −	1.279 8e−1 (2.79e−1) −	4.125 8e−3 (9.39e−5) =	**4.111 3e−3 (3.52e−5)**
	3	12	1.029 1e−1 (1.16e−2) −	2.080 7e−1 (3.36e−1) =	6.823 2e−2 (2.31e−3) −	1.534 4e−1 (8.32e−2) −	**6.149 1e−2 (2.29e−3)**
DTLZ5	2	11	4.400 7e−3 (5.39e−5) −	4.395 0e−3 (6.77e−5) −	5.082 0e−3 (1.26e−4) −	**3.980 0e−3 (9.96e−6) +**	4.094 6e−3 (2.87e−5)
	3	12	7.666 0e−3 (8.86e−4) −	6.365 3e−3 (7.02e−4) −	5.879 8e−3 (3.06e−4) −	1.436 0e−2 (9.83e−5) −	**5.358 3e−3 (4.52e−4)**
DTLZ6	2	11	4.131 8e−3 (3.00e−5) −	4.126 1e−3 (3.81e−5) −	5.691 3e−3 (2.77e−4) −	**3.966 4e−3 (6.82e−8) +**	4.073 8e−3 (2.76e−5)
	3	12	4.219 2e−3 (5.35e−5) +	**4.199 2e−3 (4.14e−5) +**	5.822 6e−3 (2.97e−4) −	1.452 9e−2 (3.89e−5) −	4.542 6e−3 (5.80e−5)
DTLZ7	2	21	**4.455 2e−3 (6.78e−5) =**	4.832 2e−2 (1.34e−1) −	5.294 5e−3 (1.72e−4) −	1.962 8e−1 (2.20e−1) −	4.466 8e−3 (5.28e−5)
	3	22	6.401 3e−2 (1.49e−3) −	1.023 8e−1 (9.48e−2) −	8.673 9e−2 (5.00e−2) −	2.165 1e−1 (7.34e−2) −	**6.266 4e−2 (1.42e−3)**
ZDT1	2	30	4.073 8e−3 (8.53e−5) −	4.187 3e−3 (8.30e−5) −	4.780 4e−3 (1.63e−4) −	1.363 3e−2 (4.32e−3) −	**3.858 6e−3 (3.47e−5)**
ZDT2	2	30	3.999 7e−3 (4.72e−5) −	4.124 0e−3 (8.26e−5) −	4.828 0e−3 (1.80e−4) −	7.695 4e−3 (1.24e−3) −	**3.859 4e−3 (3.06e−5)**
ZDT3	2	30	**4.634 9e−3 (7.96e−5) =**	4.653 6e−3 (8.15e−5) =	7.355 0e−3 (7.43e−3) −	1.760 8e−2 (5.28e−3) −	4.666 5e−3 (7.07e−5)
ZDT4	2	10	1.851 8e+1 (2.91e+0) +	2.187 7e−1 (1.99e−1) +	**5.716 7e−3 (1.43e−3) +**	1.968 9e−1 (2.01e−1) +	1.901 0e+1 (1.82e+1)
+/−/=			2/13/3	6/10/2	5/13/0	8/9/1	

表 2 给出了本文提出的 HCMOSCA 算法与其他对比算法在测试函数 DTLZ 和 ZDT 上得到的 HV 值的平均值和标准差。从表 2 可以看出，HCMOSCA 算法比其他算法有更好的性能表现。HCMOSCA 算法在 18 个测试函数中获得了 9 个最大的 HV 值，EMOSO，CMOPSO，NSGA－II，MOEA/D－DE 分别获得了 1，1，7 和 0 个最大 HV 值。从表 2 中还可以看出，部分测试问题对应的 HV 值为 0，表明对应的多目标算法在相应的测试问题上无法获得任何候选解来支配参考点。

表 2 不同对比算法在 DTLZ 和 ZDT 测试函数上的 HV 平均值和标准差

Table 2 Average and standard deviation of HV obtained by different comparative algorithms on test functions DTLZ and ZDT

Problem	M	D	EMOSO	CMOPSO	NSGA-II	MOEA/D-DE	HCMOSCA
DTLZ1	2	6	0.000 0e+0 (0.00e+0) =	3.088 4e-1 (2.71e-1) +	**5.802 9e-1 (1.10e-3) +**	5.788 1e-1 (8.08e-3) +	0.000 0e+0 (0.00e+0)
	3	7	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	**8.199 9e-1 (4.59e-3) +**	6.799 2e-1 (2.48e-1) +	0.000 0e+0 (0.00e+0)
DTLZ2	2	11	3.465 0e-1 (1.56e-4) -	3.464 4e-1 (1.47e-4) -	3.465 9e-1 (2.09e-4) -	3.470 1e-1 (3.19e-5) -	**3.472 9e-1 (8.02e-5)**
	3	12	5.429 0e-1 (2.90e-3) -	5.417 6e-1 (3.04e-3) -	5.321 7e-1 (4.74e-3) -	5.255 8e-1 (1.64e-3) -	**5.480 2e-1 (2.33e-3)**
DTLZ3	2	11	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	**3.274 4e-1 (1.36e-2) +**	1.407 6e-1 (1.58e-1) +	0.000 0e+0 (0.00e+0)
	3	12	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	**3.607 4e-1 (2.05e-1) +**	1.755 3e-1 (2.12e-1) +	0.000 0e+0 (0.00e+0)
DTLZ4	2	11	2.868 6e-1 (1.10e-1) -	2.526 9e-1 (1.25e-1) -	3.039 5e-1 (9.69e-2) -	3.468 2e-1 (5.73e-5) -	**3.471 7e-1 (9.12e-5)**
	3	12	5.322 3e-1 (4.56e-3) -	4.604 7e-1 (1.68e-1) -	5.338 7e-1 (3.65e-3) -	5.098 1e-1 (2.73e-2) -	**5.395 1e-1 (3.89e-3)**
DTLZ5	2	11	3.464 7e-1 (1.58e-4) -	3.464 2e-1 (1.62e-4) -	3.465 4e-1 (1.54e-4) -	3.470 3e-1 (3.51e-5) -	3.472 8e-1 (9.38e-5)
	3	12	1.962 1e-1 (9.06e-4) -	1.978 0e-1 (5.20e-4) -	**1.990 8e-1 (1.74e-4) =**	1.943 7e-1 (6.80e-5) -	1.989 5e-1 (2.80e-4)
DTLZ6	2	11	**3.475 5e-1 (3.75e-5) =**	3.475 5e-1 (4.26e-5) =	3.463 8e-1 (2.06e-4) -	3.472 1e-1 (7.80e-8) -	3.475 4e-1 (5.23e-5)
	3	12	2.001 8e-1 (4.18e-5) +	**2.001 8e-1 (4.03e-5) +**	1.994 3e-1 (1.61e-4) -	1.947 6e-1 (2.28e-5) -	1.999 2e-1 (4.92e-5)
DTLZ7	2	21	2.428 8e-1 (3.02e-5) -	2.361 6e-1 (2.04e-2) -	2.427 1e-1 (3.82e-5) -	2.130 8e-1 (3.32e-2) -	**2.429 5e-1 (1.01e-5)**
	3	22	2.715 8e-1 (1.54e-3) -	2.649 0e-1 (1.05e-2) -	2.663 7e-1 (5.22e-3) -	2.162 1e-1 (1.54e-2) -	**2.774 7e-1 (7.98e-4)**
ZDT1	2	30	7.196 8e-1 (1.79e-4) -	7.193 2e-1 (2.05e-4) -	7.191 3e-1 (2.34e-4) -	7.047 1e-1 (5.65e-3) -	**7.205 4e-1 (4.04e-5)**
ZDT2	2	30	4.444 7e-1 (1.42e-4) -	4.439 9e-1 (2.33e-4) -	4.439 3e-1 (2.37e-4) -	4.347 6e-1 (2.65e-3) -	**4.450 9e-1 (4.69e-5)**
ZDT3	2	30	5.995 7e-1 (1.18e-4) -	5.996 6e-1 (7.53e-5) -	**6.052 7e-1 (2.25e-2) +**	5.943 8e-1 (5.70e-3) -	5.998 0e-1 (2.55e-5)
ZDT4	2	10	0.000 0e+0 (0.00e+0) =	4.715 4e-1 (1.97e-1) +	**7.165 6e-1 (2.47e-3) +**	4.927 2e-1 (1.99e-1) +	0.000 0e+0 (0.00e+0)
+/-/=			1/11/6	3/11/4	6/11/1	5/13/0	

图 2 和图 3 绘出了 5 种多目标算法在 2 目标 DTLZ6 问题和 3 目标 DTLZ7 问题上获得的非支配解。从图 2 可以看出,HCMOSCA 在 2 目标 DTLZ6 问题上获得的非支配解具有良好的分布性,且能够很好地接近前沿面。在 2 目标 DTLZ 问题上,NSGA-II 获得的非支配解分布性相对较差。从图 3 也可以看出在 3 目标 DTLZ7 问题上,HCMOSCA 获得的非支配解有较好的分布性和收敛性。

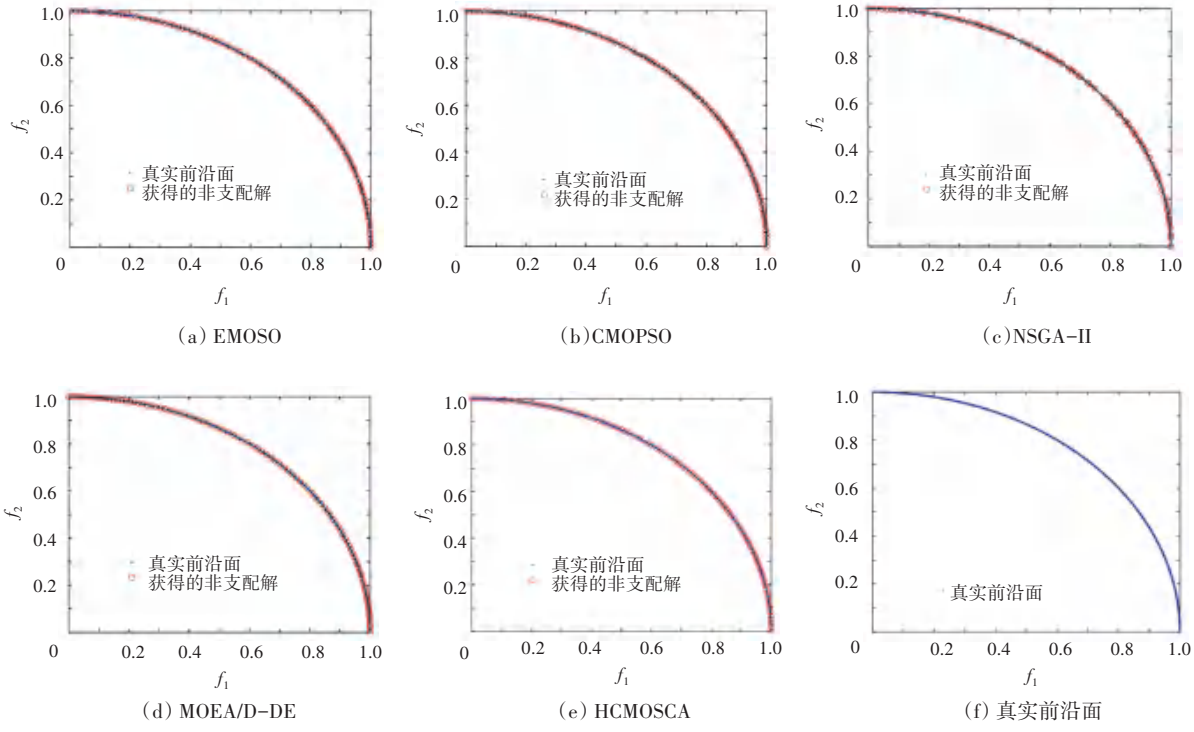


图 2 5 种多目标算法在 2 目标的 DTLZ6 问题上获得的非支配解

Fig. 2 Non-dominated solutions obtained by 5 multi-objective algorithms on the 2-objective DTLZ6 problem

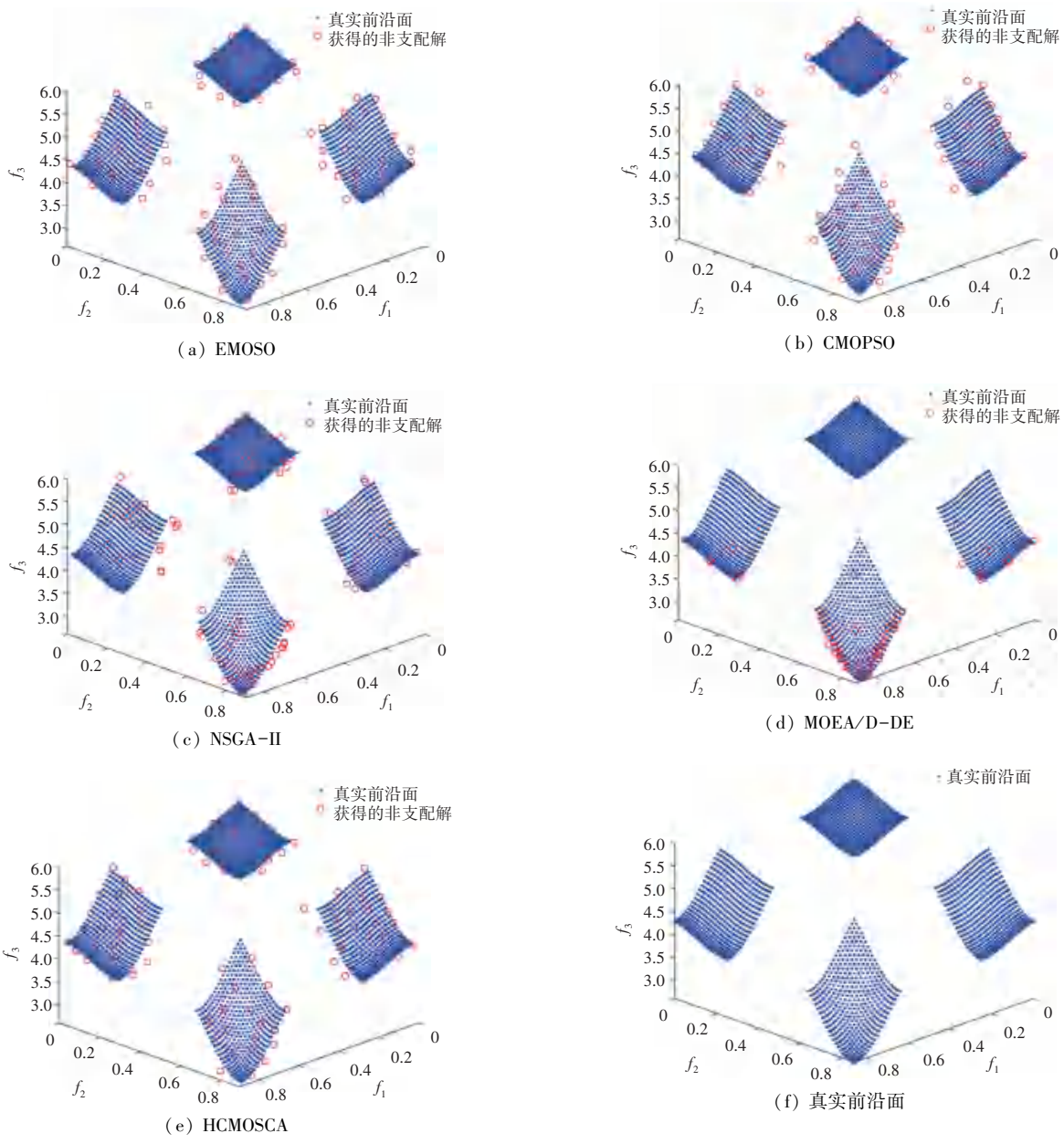


图 3　5 种多目标算法在 3 目标的 DTLZ7 问题上获得的非支配解

Fig. 3　Non-dominated solutions obtained by 5 multi-objective algorithms on the 3-objective DTLZ7 problem

4　结束语

本文基于混合距离竞争机制和角度竞争机制设计了一种多目标正余弦算法 HCMOSCA。在进化的前半阶段采用基于距离的竞争机制，侧重于种群的收敛性；在进化的后半阶段采用基于角度的竞争机制，侧重于种群的多样性。本文在 DTLZ 和 ZDT 问题集上对所提出的 HCMOSCA 算法进行了性能测试，实验结果表明，HCMOSCA 具备更优异的综合性能。未来，本文将进一步扩展 HCMOSCA，将其应用于高维多目标优化问题展开进一步研究。

参考文献

[1] LONG H, CHEN Z, HUANG H, et al. Research on Multi-objective optimization power flow of power system based on improved remora optimization algorithm[J]. Engineering Letters, 2023, 31(3): 1191-1207.

[2] GONZALEZ-SANCHEZ B, VEGA-RODRÍGUEZ M A, SANTANDER-JIMÉNEZ S. A multi-objective butterfly optimization algorithm for protein encoding[J]. Applied Soft Computing, 2023, 139: 110269.

[3] 何旺，刘敏. 采用多目标蚁狮优化算法(MALO)的环境经济调度[J]. 智能计算机与应用, 2023, 13(6): 117-121.

[4] 陆科苗，何利力. 基于改进 NSGA-II 混合算法求解多目标柔性作业车间调度问题[J]. 智能计算机与应用, 2022, 12(7):

46-51.
[5] 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.
[6] DEB K, SRINIVAS N. Multiobjective optimization using nondominated sorting in genetic algorithms [J]. Evolutionary Computation, 1995, 2(3): 221-248.
[7] ZHANG X, ZHENG X, CHENG R, et al. A competitive mechanism based multi-objective particle swarm optimizer with fast convergence[J]. Information Sciences, 2018, 427: 63-76.
[8] QIN S, SUN C, ZHANG G, et al. A modified particle swarm optimization based on decomposition with different ideal points for many-objective optimization problems[J]. Complex & Intelligent Systems, 2020, 6(2): 263-274.
[9] LUO Q, YIN S, ZHOU G, et al. Multi-objective equilibrium optimizer slime mould algorithm and its application in solving engineering problems [J]. Structural and Multidisciplinary Optimization, 2023, 66(5): 1-41.
[10] PAN J S, LIU N, CHU S C. A competitive mechanism based multi-objective differential evolution algorithm and its application in feature selection[J]. Knowledge-based Systems, 2022, 245: 108582.
[11] JIANG Q, WANG L, CHENG J, et al. Multi-objective differential evolution with dynamic covariance matrix learning for multi-objective optimization problems with variable linkages[J]. Knowledge-based Systems, 2017, 121: 111-128.
[12] MIRJALILI S. SCA: A sine cosine algorithm for solving optimization problems [J]. Knowledge-based Systems, 2016, 96: 120-133.
[13] ABUALIGAH L, DIABAT A. Advances in sine cosine algorithm: A comprehensive survey [J]. Artificial Intelligence Review, 2021, 54(4): 2567-2608.
[14] RIZK-ALLAH R M, HASSANIEN A E. A comprehensive survey on the sine-cosine optimization algorithm [J]. Artificial Intelligence Review, 2023, 56(6): 4801-4858.
[15] SCHUSSLER N, AXHAUSEN K W. SPEA2: Improving the strength pareto evolutionary algorithm [J]. Transportation Research Record, 2009, 2105(1): 28-36.
[16] ZHANG X W, LIU H, TU L P, et al. An efficient multi-objective optimization algorithm based on level swarm optimizer [J]. Mathematics and Computers in Simulation, 2020, 177: 588-602.
[17] LI H, ZHANG Q. Multiobjective optimization problems with complicated pareto sets, MOEA/ D and NSGA-II [J]. IEEE Transactions on Evolutionary Computation, 2009, 13 (2): 284-302.
[18] DEB K, THIELE L, LAUMANNS M, et al. Scalable test problems for evolutionary multiobjective optimization [M]// Evolutionary Multiobjective Optimization. 2005: 105-145.
[19] ZITZLER E, DEB K, THIELE L. Comparison of multiobjective evolutionary algorithms: empirical results [J]. Evolutionary Computation, 2000, 8(2): 173-195.
[20] COELLO C A C, CORTÉS N C. Solving multiobjective optimization problems using an artificial immune system [J]. Genetic Programming and Evolvable Machines, 2005, 6(2): 163-190.
[21] 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.
[22] TIAN Y, CHENG R, ZHANG X, et al. PlatEMO: A MATLAB Platform for evolutionary multi-objective optimization[J]. IEEE Computational Intelligence Magazine, 2017, 12(4): 73-87.
[23] TIAN Y, YANG S, ZHANG X, et al. Using PlatEMO to solve multi-objective optimization problems in applications: A case study on feature selection [C]//Proceedings of 2019 IEEE Congress on Evolutionary Computation. Piscataway, NJ: IEEE, 2019: 1710-1717.