

[文章编号] 1003—4684(2022)01-0040-06

# 基于 t-SNE 加权和高维多目标优化算法

金 涛, 朱 莉, 李 豪, 汪小豪, 姜成龙

(湖北工业大学电气与电子工程学院, 湖北 武汉 430068)

**[摘 要]** 传统的基于 t-SNE 的高维多目标优化算法在简化目标集时,虽然可以大大降低算法计算复杂度,但也可  
能损失目标集中有意义的部分属性,导致算法准确性降低。为此,对冗余目标和初始化种群两方面进行择优保留,  
提出了一种基于 t-SNE 加权和高维多目标优化算法。利用加权和对 t-SNE-NSGA II 算法处理的冗余目标集  
进行拟合,保留了部分种群的目标属性,提高了初始种群的质量,提升了算法的准确性,加快了算法收敛速度。实验  
表明,在目标超过 5 个时,基于 t-SNE 加权和高维多目标优化算法的准确性和收敛性提升明显。当目标为 10 个  
时,空间分布度提升了 38.7%。

**[关键词]** 高维多目标优化算法; 冗余目标集; t-SNESUM-NSGA II; 加权和

**[中图分类号]** TP301.6 **[文献标识码]** A

高维多目标优化问题是使多个目标在一定条件下尽可能同时达到最佳解的问题,广泛应用于科学研究、金融科技、工程设计的领域,如物资调用、人才分配、工艺设计、生产调度等<sup>[1-2]</sup>,具有重要的科研和实际价值<sup>[3]</sup>。随着目标个数的增加,传统的多目标优化算法无法很好的处理高维多目标优化问题,于是人们在传统的算法上进行改进和提升。基于 t-SNE 的高维多目标优化算法(t-SNE-NSGA II)对目标集进行分析处理,丢弃冗余目标,减少目标维度,降低计算的复杂度,提高算法的收敛性。但其在处理冗余目标集方面,只作简单的删除,使目标集中部分有用的属性可能被丢弃,导致算法的精确度降低。因此利用加权来拟合冗余目标集,进一步优化初始种群和冗余目标,从而提高算法的准确性。

## 1 问题分析

虽然 t-SNE-NSGA II 算法降低了计算的复杂度,提高算法的收敛性,但是 t-SNE-NSGA II 算法在简化目标集时,损失目标集中有意义的部分属性,导致算法准确性降低。因为在执行 t-SNE 算法后,对得到的非冗余目标集都直接进行下一次的重新初始化种群操作,导致上次算法中得到的 Pareto 解集优秀的个体没有保留。这些被舍弃的冗余目标集中存在能够反映目标集的部分属性的优秀个体,并且这些优秀个体能够引导初始化种群的方向,加快算

法的收敛速度。

针对 t-SNE-NSGA II 的不足,提出一种基于 t-SNE 加权和高维多目标优化算法(t-distributed stochastic neighbor embedding-weighted sum-nondominated sorting genetic algorithm with elite strategy, t-SNESUM-NSGA II),该算法在每次执行 t-SNE 算法后,不是直接对种群初始化,而是把经过 t-SNE 算法处理而舍去的所有冗余的目标集拟合成一个新的目标集,再把拟合好的目标集加入 t-SNE 算法得到的目标集中,最后初始化种群,这样就保留了种群中部分优秀的个体解。因此, t-SNESUM-NSGA II 算法既提高了初始种群的质量,也保留了部分种群中目标属性,提高算法的准确性,加快了算法收敛速度。

## 2 相关工作

按照问题对象的不同,高维多目标优化算法大致可以分为:不含冗余目标问题的算法和含冗余目标问题的算法<sup>[4]</sup>。第一类算法,是针对不含冗余目标问题的算法,该类算法主要可以从基于松弛 Pareto 排序方法<sup>[5]</sup>、基于聚合或分解将多目标整合为单目标问题的方法<sup>[6]</sup>和基于评价指标的方法<sup>[7]</sup>三种不同的角度来解决不含冗余目标的问题。第二类算法是针对含冗余目标问题的算法,该类算法主要可以从基于目标间互相关系的目标缩减的方法<sup>[8]</sup>与基于

[收稿日期] 2021—06—20

[基金项目] 新能源及电网装备安全监测湖北省工程研究中心开放研究基金(HBSKF202124)

[第一作者] 金 涛(1994—),男,安徽马鞍山人,湖北工业大学硕士研究生,研究方向为大数据

[通信作者] 朱 莉(1982—),女,湖北武汉人,湖北工业大学副教授,研究方向为人工智能和大数据

保持 Pareto 支配关系的目标缩减<sup>[9]</sup>的方法两种不同的角度来解决含冗余目标的问题。

随着组成问题的目标个数大幅度增加,高维多目标优化问题变得越来越复杂。不含冗余目标问题的算法,在处理多维目标时,算法搜索能力不足、计算复杂度增加,可视化难度变得越发明显。含冗余目标问题的算法,能够充分挖掘目标之间的相关性,简化目标集的个数,从而降低目标集的复杂度,获得了更广泛的应用。其中,“基于保持 Pareto 支配关系的目标缩减方法”虽然能够简化目标集,但计算目标集中非支配解集之间的支配关系会增加计算复杂度和计算机设备性能负担。

基于 t-SNE 的高维多目标优化算法主要从基于目标之间关系的目标方法的角度研究,引入 t-随机邻近嵌入(t-SNE)算法来减少高维多目标中存在的目标冗余问题。先通过仿射变换将数据点映射到概率分布上,使高维数据空间和低维数据空间的条件概率差别最小,让数据点从高维空间嵌入到低维空间中,相似度较高的距离较近,而相似度较低的距离较远。然后用 KL 散度函数作为目标函数来评估嵌入效果的优劣,最后用梯度下降法来最小化目标函数,最终得到收敛的结果。

然而 t-SNE-NSGA II 算法处理冗余目标集的方式较为粗糙,直接删除冗余目标集可能导致部分有用的属性被丢弃。因此,将冗余目标通过加权和拟合成新的目标集,从而保留目标集中的特征,增强了算法的收敛性,提高了算法的准确性。

### 3 算法设计

#### 3.1 新目标的拟合

在 t-SNE-NSGA II 算法中,每次执行 t-SNE 算法后,得到的目标集中剔除了冗余目标(图 1)。

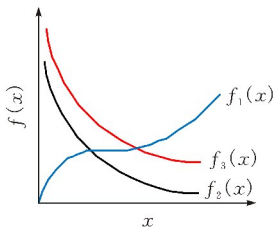


图 1  $f_1(x)$ 、 $f_2(x)$ 、 $f_3(x)$  关系分布

图 1 中  $f_2(x)$  和  $f_3(x)$  比  $f_1(x)$  和  $f_2(x)$ 、 $f_1(x)$  和  $f_3(x)$  更为相似,那么通过 t-SNE 算法计算后,  $f_2(x)$  和  $f_3(x)$  的相似度更高,是一对冗余目标,丢弃了  $f_3(x)$ 。  $f_1(x)$  和其它目标相似度最低,保留下来。

经过 t-SNE 算法丢弃的冗余目标,只和非冗余目标相似,并不是完全一致。在现实研究、工程问题

中,不会存在完全的冗余关系,如果丢弃一个相似的目标,对测试的结果会产生一定的影响。因此,在 t-SNESUM-NSGA II 算法中,考虑每次经过 t-SNE 算法中直接丢弃的目标,将这些被丢弃的冗余目标进行加权和,重新拟合成一个新的目标。因为每次 t-SNE 算法丢弃的目标与保留下来的非冗余目标具有很高的相似度,所以改进后的算法采用平均加权的方式来拟合一个新的目标。设丢弃了  $n$  个目标,则拟合后的新目标按照向量  $\{1/n, 1/n, \dots, 1/n\}$  进行加权求和。

#### 3.2 t-SNESUM-NSGA II 算法流程

传统的 t-SNE-NSGA II 算法主要的流程为 NSGA-II  $\rightarrow$  t-SNE  $\rightarrow$  NSGA-II  $\rightarrow$  t-SNE 这样的循环过程,每次进行 t-SNE 算法后都丢弃了冗余目标。t-SNESUM-NSGA II 算法对 t-SNE-NSGA II 进行改进,每次 t-SNE 算法丢弃冗余目标时,对冗余目标集进行加权求和,拟合成新的目标集并参与 NSGA-II 的优化,具体的算法流程见图 2。

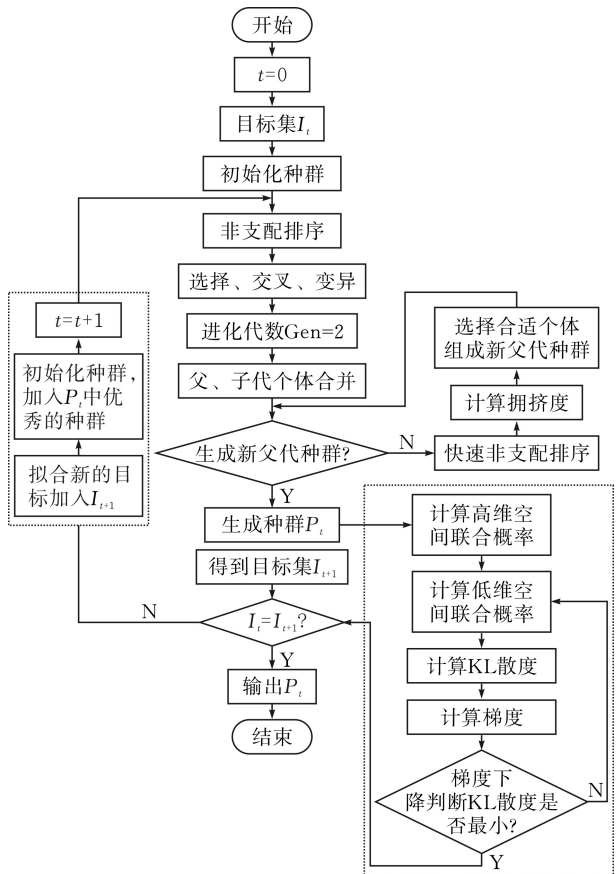


图 2 t-SNESUM-NSGA II 算法流程框图

首先,选取原始的目标集  $I_0$ ,对目标集初始化种群,然后执行 NSGA-II 算法优化种群,组成新的父代种群  $P_0$ ;再对  $P_0$  进行 t-SNE 算法优化,得到冗余目标集和非冗余目标集,对冗余目标集进行加权求和,拟合成新的目标集,与非冗余目标集合并成重组目标集;最后重复上述步骤,直到满足设置的最大

迭代次数,得到种群  $P_2, P_3, \cdots, P_{t-2}, P_{t-1}, P_t$  和目标集  $I_3, \cdots, I_t, I_{t+1}$ ,当  $I_t = I_{t+1}$ ,求出目标集的 Pareto 最优解  $P_t$ 。t-SNESUM-NSGA II 算法具体步骤见表 1。

表 1 t-SNESUM-NSGA II 算法

算法 3.1	t-SNESUM-NSGA II 算法
参数: Step1:	设 $N$ 为目标个数, $I_t$ 为目标集, $gen$ 为进化算法运行代数。
	设 $t=0$ , 初始目标集为 $I_0 = \{1, 2, \cdots, N\}$ , 对目标集进行初始化种群 $P$ , 进行选择、交叉、变异生成子代, 再通过快速非支配排序和计算拥挤度选择合适的个体组成新父代种群 $P_0$ 。
	1) 通过轮盘赌法对种群中个体进行选择
	2) 实数编码的交叉策略 (SBX)
	模拟二进制交叉:
	$x_{1j}(t) = 0.5 * [(1 + \gamma_j)x_{1j}(t) + (1 - \gamma_j)x_{2j}(t)]$
	$x_{2j}(t) = 0.5 * [(1 + \gamma_j)x_{1j}(t) + (1 - \gamma_j)x_{2j}(t)]$
	其中, $\gamma_j = \begin{cases} (2u_j)^{1/(\eta+1)} & u_j < 0.5 \\ (1/2(1-u_j))^{1/(\eta+1)} & \text{else} \end{cases}$
	3) 多项式变异
	$x_{1j}(t) = x_{1j}(t) + \Delta j$
Step2:	$\Delta j = \begin{cases} (2u_j)^{1/(\eta+1)} & u_j < 0.5 \\ (1/2(1-u_j))^{1/(\eta+1)} & \text{else} \end{cases}$
	其中, $0 \leq u_j \leq 1$
	对目标集 $I_t$ 初始化的初始种群 $P$ , 运行 NSGA-II 得到种群 $P_t$
	对 $P_t$ 对应的目标矩阵进行 t-SNE 算法优化, 得到非冗余目标集 $I'_{t+1}$ 和冗余目标集 $I''_{t+1}$ , 其中 $I_{t+1} = I'_{t+1} \cup I''_{t+1}$
	1) 计算种群 $P_0$ 的样本间两两欧氏距离, 并计出高维空间数据点对之间的联合概率 $p_{ij}$ :
	$p_{ij} = \frac{p_{j i} + p_{i j}}{2n}$
	其中, $\delta_i$ 是初始种群特征矩阵数据点 $x_i$ 为中心的高斯函数的方差, $p_{j i}$ 为种群特征矩阵数据点 $I_t$ 和数据点 $x_i$ 的相似性的条件概率 $x_j$ :
	$p_{j i} = \frac{\exp(-\ x_i - x_j\ ^2 / (2\delta_i^2))}{\sum_{k \neq i} \exp(-\ x_i - x_k\ ^2 / (2\delta_i^2))}$
	2) 计算种群 $P_0$ 特征矩阵中低维空间数据点对之间的联合概率 $q_{ij}$ :
	$q_{ij} = \frac{(1 + \ y_i - y_j\ ^2)^{-1}}{\sum_{k \neq i} (1 + \ y_i - y_k\ ^2)^{-1}}$
Step3:	3) 计算 $p_{ij}, q_{ij}$ 之间的 KL 散度, 求出目标函数 C:
	$C = KL(P \  Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}}$
	4) 求 $P$ 和 $Q$ 的梯度:
	$\frac{\partial C}{\partial y_i} = 4 \sum_j (p_{ij} - q_{ij})(y_i - y_j)(1 + \ y_i - y_j\ ^2)^{-1}$
	如果 $I_t = I_{t+1}$ , 则输出种群 $P_t$ , 即目标集的 Pareto 最优解; 否则, 直接进入 Step5
	对目标即 $I''_{t+1}$ 进行加权求和, 拟合成新的目标集, 并加入 $I_{t+1}$ 中, 拟合后的新目标按照向量 $\{1/n, 1/n, \cdots, 1/n\}$ 进行加权求和
	目标集 $I_{t+1}$ 初始化的种群 $P_{t+1}$ 和种群 $P_t$ 中 $1/n$ 种群(记为 $R_t$ )保留在一起, 即 $P_{t+1} = P_{t+1} \cup R_t$
	$t = t + 1$ , 重复 Step2- Step6, 直到满足设置的最大迭代代数, 求出目标集的 Pareto 最优解

4 实验设计与结果分析

4.1 测试函数

目前,通常选用 DTLZ 系列测试函数作为高维多目标优化算法领域的测试函数,其中包括 DTLZ1-DTLZ7。DTLZ 系列测试函数一般可以分

为含冗余目标的测试函数和不含冗余目标的测试函数。对于第一种测试函数,选择 DTLZ2 用来测试算法能否降低不含冗余目标集的目标个数。对于第二种测试函数,选择 DTLZ5 用来测试算法对简化目标的可行性。

## 4.2 测试指标

多目标优化算法中有多种性能评价指标:1)收敛性:评价解集与真正的 Pareto 最优面的趋近程度,具体的评测指标有世代距离(Generational Distance, GD);2)分布性:评价解集的多样性和均匀分布程度,具体的评测指标有空间评价方法(spacing metric, spacing)、空间分布度(diversity metric, DM)、覆盖率指标(set coverage, CS);3)综合性能:综合考虑解集的收敛性和分布性,具体的评测指标有逆世代距离(Inverted Generational Distance, IGD)、超体积指标(Hyper Volume, HV)。分别选用世代距离(GD)和空间分布度(DM)来测试 t-SNESUM-NSGA II 算法准确性和收敛性的优劣。

世代距离评价 GD 指的是解集  $P$  中的每一个点到参考集  $P'$  中的平均最小距离,表示偏离真正最优边界的程度。GD 的值越大,偏离真正最优边界越远,收敛性就越差。

空间分布度 DM 表示所获得解集的广泛程度,DM 值越小,说明解集越均匀。

## 4.3 实验参数设置

实验使用的硬件环境为 Intel Xeon(R) CPU ES-2620 v4 @ 2.10GHz, Nvidia Quadro M4000 GPU,运行内存为 32G。实验在 Matlab 仿真上实现。为了测试在不同目标个数下,算法性能的好坏,分别设置测试函数中目标个数为 3,5,10。DTLZ(I,M)表示目标在 I 维上的空间分布情况,I 为目标维度,M 为目标对象个数。具体参数如表 2 所示。

表 2 实验设置参数		
参数名称	参数表示	参数数值
种群大小	<i>populationSize</i>	500
进化代数	<i>generation</i>	10000
变异概率	<i>P<sub>m</sub></i>	0.1
交叉概率	<i>P<sub>c</sub></i>	0.9
数据维度	<i>V</i>	12
模拟二进制交叉参数	<i>l</i>	50
多项式变异参数	<i>u</i>	5

## 4.4 实验分析

采用 DTLZ2 和 DTLZ5 测试函数,在 12 维度时,分别选取目标集的个数为 3、5、10,采用 NSGA-II 算法、t-SNE-NSGA II 算法、t-SNESUM-NSGA II 算法实验测试结果,通过实验结果对比验证 t-SNESUM-NSGA II 算法的收敛性、准确性。

4.4.1 DTLZ2 (12, 3) 实验结果与分析 采取 DTLZ2 函数进行对比实验,选取 3 个目标,算法运行后的目标集与原来的目标集相同,DTLZ2(12,3)的 Pareto 前沿见图 3。说明 t-SNESUM-NSGA II 算法和 t-SNE-NSGA II 算法一样不能对不存在冗余目标的目标集进行目标降维。

4.4.2 DTLZ5 (12, 3) 实验结果与分析 采取

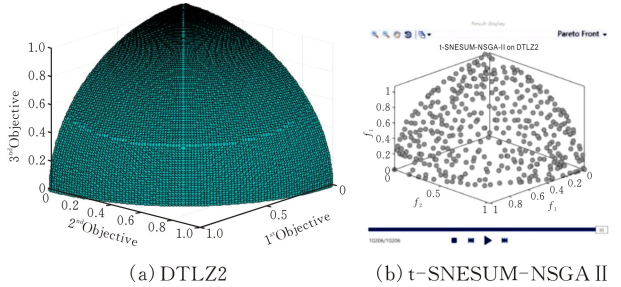


图 3 DTLZ5(12,3)的 Pareto 前沿图

DTLZ5 函数进行对比实验,选取 3 个目标,迭代次数为 10000,分别运行 NSGA-II 算法、t-SNE-NSGA II 算法和 t-SNESUM-NSGA II 算法,得到 DTLZ5 (12,3)实验结果见图 4。从图 4 可以看出 t-SNE-SUM-NSGA II 算法和 t-SNE-NSGA II 算法都具有将目标降至 2 维的能力。

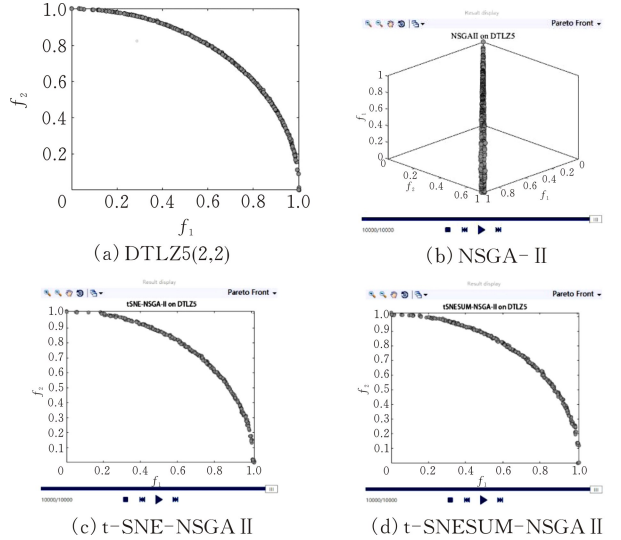


图 4 DTLZ5(12,3)实验结果图

对 DTLZ5(12,3)函数分别用 NSGA-II 算法、t-SNE-NSGA II 算法、t-SNESUM-NSGA II 算法进行测试,独立运行 10 次,测试指标取 GD 和 DM,实验结果见表 3。

表 3 DTLZ5(12,3)测试指标性能比较			
指标	NSGA-II	t-SNE-NSGA II	t-SNESUM-NSGA II
GD 标准	4.0209E-04	1.3510E-04	6.1831E-05
GD 标准偏差	3.8100E-05	6.3700E-05	2.9370E-05
DM 标准	0.62147	0.4013	0.5699
DM 标准偏差	0.194	0.234	0.0133

由表 3 中的数据可知,t-SNESUM-NSGA II 算法的 GD 和 DM 两个指标的分别为:6.1831E-05, 0.5699;t-SNE-NSGA II 算法的 GD 和 DM 两个指标的分别为: 1.3510E-04, 0.4013。由于 t-SNE-SUM-NSGA II 算法的 GD 比 t-SNE-NSGA II 算法和 NSGA-II 算法 GD 都小,表示 t-SNESUM-NSGA II 算法偏离真正最优边界最小,收敛性最好;而 t-SNESUM-NSGA II 算法 DM 比 NSGA-II 算法



小,但是大于 t-SNE-NSGA II 算法的 DM,表示 t-SNESUM-NSGA II 算法的解集广泛程度介于两个算法之间。

综上所述,t-SNESUM-NSGA II 算法收敛度比 NSGA-II 算法和 t-SNE-NSGA II 算法更好,解集分布度相对 NSGA-II 算法提高了,但是对于 t-SNE-NSGA II 算法来说却降低了。t-SNESUM-NSGA II 算法收敛速度得到了加强,但是在解集的分布来说反而降低了,这说明 t-SNESUM-NSGA II 算法虽然得到的解集具有更好的分布度,但是在目标集低的时候解集分布度还是低于 t-SNE-NSGA II 算法,猜想当目标个数较低时,t-SNESUM-NSGA II 算法优势不够明显。

**4.4.3 DTLZ5 (12, 5) 实验结果与分析** 采取 DTLZ5 函数进行对比实验,选取 5 个目标个数,分别运行 t-SNE-NSGA II 算法和 t-SNESUM-NSGA II 算法,得到的 DTLZ5(12,5)收敛速度见图 5。由上面的结论可以知道,t-SNE-NSGA II 算法和 t-SNESUM-NSGA II 算法都可以把目标降至 2 维,但从图 5 可以看出,t-SNE-NSGA II 算法大概在迭代次数为 6800 代左右时完全收敛,而 t-SNESUM-NSGA II 算法在 6400 代左右就可以完全收敛,这说明 t-SNESUM-NSGA II 算法的收敛速度优于 t-SNE-NSGA II 算法的收敛速度。

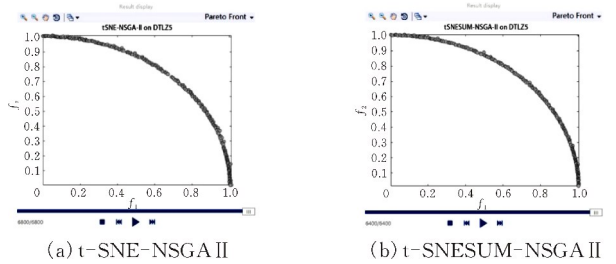


图 5 DTLZ5(12,5)收敛速度图

对 DTLZ5(12,5)函数分别用 NSGA-II 算法、t-SNE-NSGA II 算法、t-SNESUM-NSGA II 算法,独立运行 10 次,测试指标取 GD 和 DM,实验结果见表 4。

表 4 DTLZ5(12,5)测试指标性能比较

指标	NSGA-II	t-SNE-NSGA II	t-SNESUM-NSGA II
GD 标准	5.7065E-02	2.6509E-05	1.6389E-03
GD 标准偏差	2.1500E-03	4.9200E-05	1.0791E-05
DM 标准	0.35211	0.33982	0.3519
DM 标准偏差	0.00977	0.0103	0.0165

由表 4 数据可知,t-SNESUM-NSGA II 算法的 GD 和 DM 两个指标分别为:1.6389E-03,0.3519;t-SNE-NSGA II 算法的 GD 和 DM 两个指标的分别为:2.6509E-05,0.33982。由于 t-SNESUM-NSGA

II 算法的 GD 比 NSGA-II 算法和 t-SNE-NSGA II 算法的 GD 都小,表示 t-SNESUM-NSGA II 算法偏离真正最优边界最小,收敛性最好;t-SNESUM-NSGA II 算法 DM 比 NSGA-II 算法和 t-SNE-NSGA II 算法的 DM 也都小,表示 t-SNE-NSGA II 算法的解集广泛程度最小。

因此,t-SNESUM-NSGA II 算法收敛度比 NSGA-II 算法和 t-SNE-NSGA II 算法更好,解集分布度相对 NSGA-II 算法提高了,但是对于 t-SNE-NSGA II 算法来说仍然没有提升。t-SNESUM-NSGA II 算法在收敛速度得到加强,但是在解集的分布来说反而降低了,这说明 t-SNESUM-NSGA II 算法虽然得到的解集具有更好的分布度,但是在目标集低的时候解集分布度还是低于 t-SNE-NSGA II 算法。

**4.4.4 DTLZ5 (12, 10) 实验结果与分析** 采取 DTLZ5 函数进行对比实验,选取 10 个目标个数,迭代次数为 10000 次,分别运行 t-SNE-NSGA II 算法和 t-SNESUM-NSGA II 算法,得到的 DTLZ5(12, 10)空间分布见图 6,从图可以看出 t-SNESUM-NSGA II 算法性能明显优于 t-SNE-NSGA II 算法。

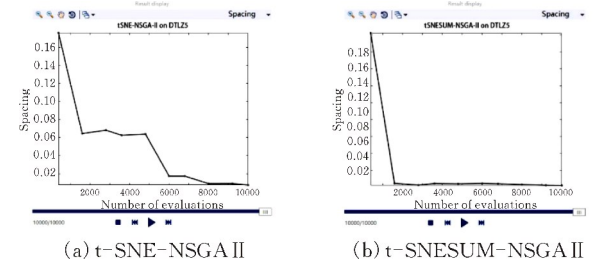


图 6 DTLZ5(12,10)空间分布图

对 DTLZ5(12,10)函数分别用 NSGA-II 算法、t-SNE-NSGA II 算法、t-SNESUM-NSGA II 算法进行测试,独立运行 10 次,测试指标取 GD 和 DM,实验结果见表 5。

表 5 DTLZ5(12,10)测试指标性能比较

指标	NSGA-II	t-SNE-NSGA II	t-SNESUM-NSGA II
GD 标准	3.5564E-02	1.7509E-04	7.2521E-06
GD 标准偏差	1.0300E-03	9.3200E-05	5.8145E-07
DM 标准	0.36276	0.2963	0.2137
DM 标准偏差	0.0227	0.0308	0.0112

由表 5 中的数据可知,t-SNESUM-NSGA II 算法的 GD 和 DM 两个指标分别为:7.2521E-06,0.2137;t-SNE-NSGA II 算法的 GD 和 DM 两个指标分别为:1.7509E-04,0.2963。由于 t-SNE-NSGA II 算法的 GD 比 NSGA-II 算法和 t-SNE-NSGA II 算法的 GD 都小,表示 t-SNESUM-NSGA II 算法偏离真正最优边界最小,收敛性最好;t-SNESUM-NSGA II 算法的 DM 比 NSGA-II 算法和 t-SNE-

NSGA II 算法的 DM 也都小,表示 t-SNE-NSGA II 算法的解集广泛程度最小。

综上所述,t-SNESUM-NSGA II 算法无论是收敛到最优边界能力还是解集分布度都明显优于 t-SNE-NSGA II 算法,验证了随着目标个数的增加,t-SNESUM-NSGA II 算法性能比 t-SNE-NSGA II 算法的更好。

## 5 结论

t-SNESUM-NSGA II 算法在 t-SNE-NSGA II 算法的基础上进行改进,进一步优化初始种群和冗余目标,提高了算法的准确性。该算法重点在 t-SNE 分析处理得到的冗余目标集进行拟合形成了新的目标集,并加入下一步的运算中;而且在每次初始化种群的时候,保留父代种群的部分个体。虽然 t-SNESUM-NSGA II 算法中采用对冗余目标集直接加权,拟合成新的目标,这样虽然可以保留目标集的部分属性,但大部分属性仍旧丢失。因此在面对冗余目标集时候,如何选择一个好的拟合方法,怎样分配拟合后各个目标集间的权重,来充分挖掘目标集的特征,降低目标维度,还有待进一步研究。

### [ 参 考 文 献 ]

[1] RAJALAKSHMI J AND DURAIRAJ S. Application of multi-objective optimization algorithm for siting and sizing of distributed generations in distribution net-

works[J]. Journal of Combinatorial Optimization, 2020, 41(2): 267-289.

[2] HAN W, DENG Q, GONG G, et al. Multi-objective evolutionary algorithms with heuristic decoding for hybrid flow shop scheduling problem with worker constraint[J]. Expert Systems with Applications, 2020, 168(12):114282.

[3] 郭梅,李西兴. 基于改进遗传算法的多目标优化生产调度模型[J]. 数字制造科学, 2018(1): 64-70.

[4] 过晓芳. 高维多目标优化算法研究综述[J]. 科技视界, 2015(15): 21-22.

[5] HADKA D, REED P. Borg: An auto-adaptive many-objective evolutionary computing framework[J]. Evolutionary Computation, 2013, 21(2): 231-259.

[6] FELLOW, JAIN H, et al. An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, Part II: handling constraints and extending to an adaptive approach [J]. IEEE Transactions on Evolutionary Computation, 2014, 18(4): 602-622.

[7] ZHANG X, TIAN Y, JIN Y. A knee point-driven evolutionary algorithm for many-objective optimization [J]. IEEE Transactions on Evolutionary Computation, 2015, 19(6): 761-776.

[8] 刘琼,熊书平,湛梦梦. 基于改进精英策略的 PCA-NSGA II 的高维目标调度优化[J]. 计算机集成制造系统, 2020, 26(9): 2474-2483.

[9] BROCKHOFF, DIMO, DERBEL, et al. Feature Cluster on "Evolutionary multiobjective optimization" Preface[J]. European Journal of Operational Research, 2015, 243(2): 345-346.

# High-dimensional Multi-objective Optimization Algorithm Based on t-SNE Weighted Sum

JIN Tao, ZHU Li, LI Hao, WANG Xiaohao, JIANG Chenglong

(School of Electrical and Electronic Engineering, Hubei Univ. of Tech. Wuhan 430068, China)

**Abstract:** When the traditional high-dimensional multi-objective optimization algorithm based on t-distributed stochastic neighbor embedding (t-SNE-NSGA II) simplifies the target set, although it can greatly reduce the computational complexity of the algorithm, it may also lose some meaningful attributes of the target set, resulting in the accuracy of the algorithm decreased. Therefore, a high-dimensional multi-objective optimization algorithm based on t-SNE weighted sum was proposed to preserve redundant target and initial population. The weighted sum is used to fit the redundant target set processed by t-SNE-NSGA II algorithm, and the target attributes of part of the population are retained, the quality of the initial population is improved, the accuracy of the algorithm is improved, and the convergence speed of the algorithm is accelerated. Experimental results show that when the number of targets is more than 5, the accuracy and convergence of the high-dimensional multi-objective optimization algorithm based on t-SNE weighted sum are improved obviously. When the number of targets was 10, the spatial distribution increased by 38.7%.

**Keywords:** High-dimensional multi-objective optimization algorithm; redundant objective set; t-SNESUM-NSGA II; weighted summation

[责任编辑: 张岩芳]