

# 基于网格投影的超多目标进化算法

高智阔<sup>1,2</sup>, 陈未如<sup>1,2</sup>, 彭弗楠<sup>1,2</sup>

(1. 沈阳化工大学 计算机科学与技术学院, 辽宁 沈阳 110142;

2. 辽宁省化工过程工业智能化技术重点实验室, 辽宁 沈阳 110142)

**摘要:**针对超多目标优化问题求解困难的问题,研究如何得到收敛性和分布性较优的解集,提出了一种基于网格投影的超多目标进化算法-GPEA。该算法根据决策需求将超多目标优化问题的目标空间进行分解,得到投影维目标空间和自由维目标空间;再将投影维目标空间分割为若干投影格,将自由维目标空间分段成若干自由格。算法在每个投影格上进行种群进化,并根据个体相对投影格的位置采用两测度策略筛选个体。第一测度是对落入到投影格内的个体使用非支配排序和自由维目标空间个体筛选策略,选择收敛性和分布性较优的个体作为候选种群。当落入到投影格内的个体数量不足时,进行第二测度筛选,根据个体相对投影格的距离排队,选择相对较近的个体并入到候选种群中。分析了算法的性能,通过对标准测试函数在不同目标下的求解,实验证明基于网格投影的超多目标进化算法能够有效地求解超多目标优化问题。

**关键词:**超多目标优化;进化算法;目标空间;网格;投影;评价指标

中图分类号:TP301.6

文献标识码:A

文章编号:1673-629X(2023)05-0022-07

doi:10.3969/j.issn.1673-629X.2023.05.004

## Many-objective Evolutionary Algorithm Based on Grid Projection

GAO Zhi-kuo<sup>1,2</sup>, CHEN Wei-ru<sup>1,2</sup>, PENG Fu-nan<sup>1,2</sup>

(1. School of Computer Science and Technology, Shenyang University of Chemical Technology,  
Shenyang 110142, China;

2. Liaoning Province Key Laboratory of Industrial Intelligence Technology on Chemical Process,  
Shenyang 110142, China)

**Abstract:** Aiming at the difficult problem of solving many-objective optimization problems, how to obtain a solution set with better convergence and distribution is studied, and a many-objective evolutionary algorithm based on grid projection (GPEA) is proposed. According to the decision-making requirements, the objective space is decomposed into the projection-dimensional space and the free-dimensional space, then the projected dimension space is divided into several projection grids, and the free dimension space is segmented into several free grids. The GPEA performs population evolution on each projected grid, and uses a two-measure strategy to select individuals according to their position relative to the projected grid. The first measure is to use nondominated sorting and a free-dimensional space individual selecting strategy for those falling into the projection grid, and the individuals with better convergence and distribution are selected as a candidate population. When the number of individuals falling into the projected grid is insufficient, the second measure is performed, and the individuals relatively close to the projected grid are selected to be incorporated into the candidate population. The performance of the algorithm is analyzed. By solving the standard test function under different objectives, the experiment proves that the GPEA can effectively solve the many-objective optimization problem.

**Key words:** many-objective optimization; evolutionary algorithm; objective space; grid; projection; performance indicators

## 0 引言

在社会生产和工程应用等领域,存在着许多的优化问题涉及对多个目标进行优化,并且绝大多数目标之间是相互关联并相互冲突的。当目标数为2或3

时,这类问题被称为多目标优化问题(multi-objective optimization problem, MOP)<sup>[1]</sup>。当目标数大于3时,这类问题被称为超多目标优化问题(many-objective optimization problem, MaOP)<sup>[2]</sup>。社会实际应用中出

收稿日期:2022-07-20

修回日期:2022-11-22

基金项目:辽宁省自然科学基金项目(1645756237146);辽宁省教育厅科研项目(LJ2020024)

作者简介:高智阔(1998-),男,研究生,研究方向为超多目标优化问题和进化算法;陈未如(1963-),男,教授,研究方向为智能计算和数据挖掘;通讯作者:彭弗楠(1979-),女,讲师,研究方向为智能计算和无线传感器网络。

现了大量的 MaOP 例子,如路径规划问题<sup>[3]</sup>、电力优化问题<sup>[4]</sup>、零件加工问题<sup>[5]</sup>等。

针对多目标优化问题,研究人员结合进化算法提出了很多多目标进化算法(multi-objective evolutionary algorithms, MOEAs),根据进化机制<sup>[6]</sup>不同可将 MOEAs 分为以下三种:基于支配关系的 NSGA-II<sup>[7]</sup>、基于分解的 MOEA/D<sup>[8]</sup>、基于性能评价指标的 IBEA<sup>[9]</sup>。当目标数量过多时,使用这些算法求解超多目标优化问题的性能降低,在超多目标空间中保持解集的分布性和收敛性也很困难<sup>[10]</sup>。研究人员针对以上问题提出了很多超多目标进化算法(many-objective evolutionary algorithms, MaOEAs):如 NSGA-III<sup>[11]</sup>在 NSGA-II 的基础上,引入了参考点针对临界支配层求解个体;R2-EMOA<sup>[12]</sup>针对种群采用快速非支配排序方法,针对临界层使用 R2 指标筛选个体;MOEA/DP<sup>[13]</sup>结合 MOEA/D 思想,针对投影面分解求解个体;KnRVEA<sup>[14]</sup>为子种群引入拐点自适应策略和参考向量相关联协同进化每一个子种群;NSGA/P<sup>[15]</sup>结合 MOEA/P<sup>[16]</sup>的投影面思想,对 NSGA-II 进行投影支配改进;SPEA/R<sup>[17]</sup>采用目标空间分解为参考向量的策略,使用多样性优先收敛性第二的选择策略筛选个体。

随着目标维数的增加,种群内的个体大都是互不支配的。使用基于支配关系的多目标进化算法和超多目标进化算法求解个体会存在解集收敛性和分布性不足的情况,如 NSGA-II 和 NSGA-III。为解决上述问题,该文提出了一种基于网格投影的超多目标进化算法。该算法采用 MOEA/P 算法的投影思想针对目标空间进行降维操作,将目标空间划分为投影维目标空间和自由维目标空间,降低了对超多目标优化问题的求解难度,针对自由维目标空间引入网格适应度选取策略<sup>[18]</sup>,使得算法求得的解集具有良好的收敛性和分布性。

## 1 背景知识

### 1.1 基本概念

多目标优化问题在一般情况下可以描述为最小化问题,数学描述如下:

$$\text{Minimize } F(x) = (F_1(x), F_2(x), \dots, F_M(x)) \quad (1)$$

$$\text{Subject to } x \in \Omega$$

$$\Omega = \{x \in R^n \mid g_j(x) \geq 0, h_k(x) = 0 \quad (j = 1, 2, \dots, J; k = 1, 2, \dots, K)\} \quad (2)$$

其中,  $F_i (i = 1, 2, \dots, M)$  是需要最小化的第  $i$  个目标;  $\Omega$  是决策空间。在公式(2)中,  $g$  是包含  $J$  个等式和不等式的约束函数,  $h$  是包含  $K$  个等式的约束函数;  $x = (x_1, x_2, \dots, x_n)$  是决策空间中的决策向量;当目标个数  $M$  的值大于 3 时,称其为超多目标优化问题。

定义 1 Pareto 支配:针对决策空间中的两个个体  $x$  和  $y$ ,有  $x$  支配  $y$  (记作  $x < y$ ),当且仅当  $x$  和  $y$  满足:

$$\begin{aligned} \forall i \in (1, 2, \dots, M): F_i(x) &\leq F_i(y) \wedge \\ \exists j \in (1, 2, \dots, M): F_j(x) &< F_j(y) \end{aligned} \quad (3)$$

定义 2 Pareto 最优解:针对决策空间中的一个个体  $x^*$ ,当且仅当  $x^*$  不被该决策空间中的其他任何个体  $x$  支配时,则称  $x^*$  为 Pareto 最优解。

定义 3 Pareto 最优解集:由 Pareto 最优解组成的集合称为 Pareto 最优解集,即:

$$P^* = \{x^*\} = \{x \in \Omega \mid \neg \exists x' \in \Omega, x' < x\} \quad (4)$$

定义 4 Pareto 前沿:  $P^*$  中的全部个体映射到目标空间的集合,称为 Pareto 前沿,即:

$$\begin{aligned} PF^* = \{F(x) = (F_1(x), F_2(x), \dots, F_M(x)) \mid \\ x \in P^*\} \end{aligned} \quad (5)$$

### 1.2 MOEA/P 算法框架

MOEA/P 算法采用投影的思想,根据决策者的需求将目标空间分解为两部分,分别是投影面和自由维,以三维目标问题为例,如图 1 所示。其中投影面是决策者主要侧重的目标集,而自由维的目标集则在划分投影面的基础上进一步求解。

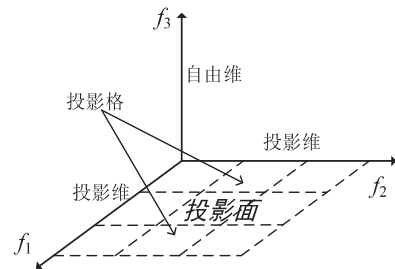


图 1 将目标空间分解为投影面和自由维

### 1.3 GrEA 算法思想

GrEA 算法采用以个体为中心计算的思想,将个体在目标空间中的目标值计算取代为网格坐标值计算,将网格排序 (GR)、网格拥挤距离 (GCD) 和网格坐标点距离 (GCPD) 作为个体筛选条件,筛选出目标空间中收敛性和分布性较优的个体。

## 2 GPEA 算法

受 MOEA/P 算法和 GrEA 算法的启发,该文针对投影维目标空间引入 MOEA/P 算法的投影思想,针对自由维目标空间引入 GrEA 算法中的网格适应度策略筛选个体,设计了一种基于网格投影的超多目标进化算法 (GPEA)。

### 2.1 算法基本思想

GPEA 算法的基本思想是使用 MOEA/P 算法框架,将目标空间分解成投影维目标空间和自由维目标空间两部分。其中投影维目标空间被划分成若干个投

影格。求解过程中,针对各投影格分别进化。在每代进化中,筛选落入投影格内的个体,再根据 GrEA 算法的网格思想将自由维目标空间均匀分段成若干自由格,计算个体相对于自由格的空间属性,利用个体非支配排序结果和自由维目标空间个体筛选策略对这些个体进行综合筛选。

### 2.1.1 自由维目标空间个体筛选策略

自由维目标空间个体选择策略是以个体在自由维目标空间内的自由格排序(FGR)作为首要筛选条件,自由格拥挤距离(FGCD)作为次要筛选条件,自由格坐标点距离(FGCPD)作为最后的筛选条件。

在对第  $i$  个自由维目标分配自由格坐标时,自由格下界、上界对应的目标值  $lb_i$  和  $ub_i$  见公式(6)和公式(7)。

$$lb_i = \min_i(P) - (\max_i(P) - \min_i(P))/(2 \times g) \quad (6)$$

$$ub_i = \max_i(P) + (\max_i(P) - \min_i(P))/(2 \times g) \quad (7)$$

其中,  $\min_i(P)$  和  $\max_i(P)$  是种群  $P$  在第  $i$  个自由维目标中的最小值和最大值,  $g$  是对自由维目标空间中各目标的分段数。

自由格在第  $i$  个自由维目标中的长度为  $d_i$ , 计算方法见公式(8)。

$$d_i = (ub_i - lb_i)/g \quad (8)$$

这样,每一自由维的目标都被均匀分割成  $g$  个分段,所有自由维上的各个分段组合成自由格。如果把每个自由维各分段标号为  $0, 1, \dots, g-1$ , 用这些分段标号作为个体的自由格坐标。

定义5 自由格坐标:个体  $x$  所落入的自由格的坐标。个体  $x$  在第  $i$  个自由维上的自由格坐标是该个体所落入自由格在第  $i$  自由维上的分段标号。个体  $x$  的自由格坐标计算方法见公式(9)。

$$FG_i(x) = \lfloor (FF_i(x) - lb_i)/d_i \rfloor \quad (9)$$

其中,  $lb_i$  是自由格下界,  $d_i$  是自由格长度,它们的计算方法见式(6)和式(8),  $FF(x) = (FF_1(x), FF_2(x), \dots, FF_f(x))$  为个体  $x$  在  $f$  个自由维组成的空间中对应的各自由维坐标向量,  $FF_i(x)$  则是该向量在第  $i$  自由维上的分量。

定义6 自由格距离:自由格距离 FGD 用于表示个体  $x$  和  $y$  在自由维目标空间中的自由格坐标的相对位置关系,自由格距离 FGD 由公式(10)给出。

$$FGD(x, y) = \sum_{i=1}^f |FG_i(x) - FG_i(y)| \quad (10)$$

当个体  $x$  和  $y$  的自由格距离值小于自由维数  $f$  时,将个体  $y$  作为个体  $x$  的邻居。在自由维目标空间中,个体  $x$  的邻居组成的集合为  $FN(x)$ 。

定义7 自由格排序:将个体在各个自由维的自由格坐标值的总和作为个体的自由格排序值。公式(11)为个体在自由维目标空间中的自由格排序计算公式。

$$FGR(x) = \sum_{i=1}^f FG_i(x) \quad (11)$$

定义8 自由格拥挤距离:将个体  $x$  在自由维目标空间上与其所有邻居之间的距离作为个体  $x$  自由格拥挤距离,其中  $N(x)$  是个体  $x$  的邻居组成的集合,  $f$  是自由维数。公式(12)为个体在自由维目标空间中的自由格拥挤距离计算公式。

$$FGCD(x) = \sum_{y \in FN(x)} [f - FGD(x, y)] \quad (12)$$

定义9 自由格坐标点距离:将个体在自由维目标空间上的自由维目标值与自由格边界点目标值的欧氏距离作为自由格坐标点距离。公式(13)为个体在自由维目标空间中的自由格坐标点距离计算公式。

$$FGCPD(x) =$$

$$\sqrt{\sum_{i=1}^f \{ [FF_i(x) - (lb_i + FG_i(x) \times d_i)] / d_i \}^2} \quad (13)$$

### 2.1.2 投影格适应度

投影格适应度是指个体  $x$  相对于投影格中心点  $Z$  的位置关系,个体的投影格适应度计算方法由公式(14)给出。

$$GP(x) =$$

$$\begin{cases} 0, & \max_{1 \leq i \leq k} \{ |FP_i(x) - Z_i| \} \leq \frac{1}{2k} \\ \sqrt{\sum_{i=1}^p [FP_i(x) - Z_i]^2}, & \max_{1 \leq i \leq k} \{ |FP_i(x) - Z_i| \} > \frac{1}{2k} \end{cases} \quad (14)$$

其中,投影格中心点向量  $Z = (Z_1, Z_2, \dots, Z_p)$ ;  $FP$  为归一化后的投影维目标空间,对应个体  $x$  在该空间的归一量为  $FP(x) = (FP_1(x), FP_2(x), \dots, FP_p(x))$ ;  $p$  为投影维目标空间目标数,  $k$  为投影维分段数。

## 2.2 算法框架

GPEA 算法框架描述如下:

GPEA 算法

输入:

$M$  (目标个数),

$N$  (种群大小),

$E$  (最大进化代数),

DS (标决策空间),

$g$  (自由维自由格分段数),

$k$  (投影维投影格分段数)

输出:目标解集 OP

过程:

步骤 1:目标空间划分。

根据 DS 设置将目标空间划分为投影维目标空间和自由维



目标空间,其中投影维目标空间目标数为 $p$ ,自由维目标空间目标数为 $f$ ;投影维目标空间划分投影格数为 $k^p$ ,自由维目标空间的自由格数为 $g^f$ ;划分投影格并为投影格分配投影格序号 $i(i = 1, 2, \dots, k^p)$ 。

步骤2:初始化种群。

设投影格序号 $i = 1$ ,为其初始化大小为 $N$ 的种群 $P_i$ ;

步骤3:种群进化。

步骤3.1:对种群 $P_i$ 内的个体进行交叉变异产生子代个体,合并父代和子代的个体组成新种群 $CP_i$ ;计算种群 $CP_i$ 内所有个体的目标函数值,对投影维目标空间进行归一化操作,为所有个体计算投影格适应度;

步骤3.2:从种群 $CP_i$ 中选择 $N$ 个良好的个体。

将种群 $CP_i$ 内全部个体投影到投影维目标空间中按照个体的投影格适应度进行分类,将落入投影格内的个体放入列表PL中,将落入投影格外的个体放入列表FL中;如果列表PL内的个体数大于等于 $N$ ,则执行步骤3.2.1,否则执行步骤3.2.2;

步骤3.2.1:针对列表PL内的个体在自由维目标空间中进行非支配排序,生成的 $R$ 个非支配子集 $F_1, F_2, \dots, F_R$ ;将非支配子集内个体依次放入列表LN中并保证LN内个体数小于 $N$ ,直到 $F_r$ ,当 $F_r$ 放入列表LN时,LN内个体数刚好大于等于 $N$ ;计算非支配子集 $F_r$ 中个体的自由格排序FGR、自由格拥挤距离FGCD、自由格坐标点距离FGCPD,利用自由维目标空间个体筛选策略依次选择较优的个体放入列表LN中,直至列表LN内的个体数正好等于 $N$ 。转步骤3.3;

步骤3.2.2:将列表PL内的所有个体放入到列表LN中,依次选择列表FL内投影适应度较优的个体依次放入到列表LN中,直至列表LN内的个体数量正好等于 $N$ 。转步骤3.3;

步骤3.3:判断投影格种群 $P_i$ 是否达到了最大进化代数 $E$ ;

若种群 $P_i$ 未达到最大进化代数,则将列表LN中的个体作为投影格 $i$ 的新一代种群 $P_i$ ,继续执行步骤3的种群进化操作;

若种群 $P_i$ 达到了最大进化代数,则将列表LN中的个体并入目标解集OP中,并在OP中只保留非支配个体。此时,若 $i < k^p$ ,即投影格 $i$ 不是最后一个投影格,则将列表LN中的个体作为下一个投影格 $i+1$ 的初始种群 $P_{i+1}$ ,并令 $i = i+1$ ,继续执行步骤3进行下一投影格的进化操作;否则结束进化转步骤4。

步骤4:输出OP。

### 3 实验与分析

#### 3.1 测试问题

该文选取DTLZ<sup>[19]</sup>测试问题集中的DTLZ1~DTLZ4测试问题作为比较的基础,其中DTLZ1和DTLZ3测试函数为算法收敛到Pareto前沿创造了很多困难,DTLZ2和DTLZ4测试函数用于测试算法处理不同形状问题的能力。这些测试问题都可以扩展到任意个数的目标和决策向量,用来验证所提算法的性能。

#### 3.2 性能指标

为了评价算法的综合性能,采用了反向迭代距离指标(IGD)<sup>[20]</sup>来评价算法求得解集的收敛性和分布

性。IGD衡量的是算法求得的解集与真实Pareto前沿的个体之间的最小距离的平均值,计算IGD需要预先得到该问题的一组均匀的Pareto前沿真实解集。

IGD指标的计算公式为:

$$IGD(S, P^*) = \frac{\sum_{i=1}^{|P^*|} \text{dist}(x, s)}{|P^*|} \quad (15)$$

$$\text{dist}(x, s) = \sqrt{\sum_{i=1}^M (s_i - x_i)^2} \quad (16)$$

其中, $S$ 为算法求得的一组Pareto近似解集; $P^*$ 为一组均匀采样的Pareto前沿点集; $x$ 是 $P^*$ 中的个体; $s$ 是 $S$ 中的个体; $|P^*|$ 是Pareto前沿点集中个体的数量; $s_i$ 是Pareto近似解集中个体 $s$ 在第 $i$ 个目标中的目标值; $x_i$ 是Pareto前沿点集中个体 $x$ 在第 $i$ 个目标中的目标值; $\text{dist}(x, s)$ 是个体 $x$ 到 $S$ 最近的个体的欧氏距离。IGD的值越小,算法求得的解集越接近Pareto真实前沿,表明算法求得的解集具有较好的收敛性和多样性。

#### 3.3 实验设计

为了验证该算法的性能,实验选取MOEA/D、GrEA、NSGA-III、NSGA/P、MOEA/DP作为对比,在DTLZ1~DTLZ4测试问题的3、5、7、10目标上进行实验。

交叉变异参数设置:为所有算法使用模拟二进制交叉操作和多项式变异操作,其中交叉概率设置为1,变异概率设置为 $1/n$ , $n$ 是决策变量个数,交叉变异的分布指标都设置为20。

算法自身对比参数设置:在10目标的DTLZ1~DTLZ4测试问题,决策变量个数 $n = 12$ ,进化代数 $E = 6\,000$ ,种群大小 $N = 220$ ,性能指标选用IGD的实验条件下进行算法对比参数设置。为探究投影维目标空间设置对算法性能的影响,设置投影维目标空间数 $p$ 依次为1,2,...,7,投影维分段数 $k = 2$ ,自由维分段数 $g = 8$ ;为探究投影维分段数 $k$ 对算法性能的影响,设置 $k$ 的值依次为1,2,...,7,自由维分段数 $g = 8$ ;为探究自由维分段数 $g$ 对算法的影响,设置 $g$ 的值依次为4,6,8,10,12,14,16,投影维分段数 $k = 2$ 。以上每组测试独立运行20次。

不同算法对比参数设置:设置所有测试问题对应的投影维分段数 $k = 2$ ,自由维分段数 $g = 8$ ;针对3目标问题,决策变量个数 $n = 6$ ,种群大小 $N = 190$ ,进化代数 $E = 3\,000$ ,投影维数 $p = 1$ ;针对5目标问题,决策变量个数 $n = 8$ ,种群大小 $N = 210$ ,进化代数 $E = 4\,000$ ,投影维数 $p = 2$ ;针对7目标问题,决策变量个数 $n = 10$ ,种群大小 $N = 210$ ,进化代数 $E = 5\,000$ ,投影维数 $p = 3$ ;针对10目标问题,决策变量个数 $n = 12$ ,种群

大小  $N = 220$ , 进化代数  $E = 6\,000$ , 投影维数  $p = 4$ ; 每组测试独立运行 30 次。

### 3.4 结果与分析

图 2 表示在投影维分段数  $k$  和自由维分段数  $g$  不变的情况下, 投影维数  $p$  变化对 IGD 值变化的曲线。从图中可以看出, 当投影维数小于 3 时, IGD 值逐渐减小, 使用投影思想可以提高算法的求解效果, 降低了算法在自由维目标空间中求解难度; 但是在投影维数超过 3 以后, IGD 的值开始缓慢增加, 在投影维数过多的时候, 个体在自由维目标空间中的收敛性和分布性相对片面地表示个体在目标空间中的收敛性和分布性, 此时算法的求解效果较差; 当投影维数约占目标总数的  $1/3$  时, 算法的求解效果较好。

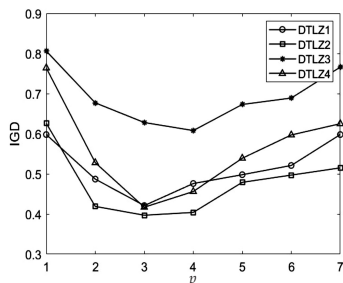


图 2 GPEA 在 DTLZ 测试问题上不同投影维数的 IGD 变化曲线

图 3 表示在投影维数  $p$  和自由维分段数  $g$  不变的情况下, 投影维分段数  $k$  变化对 IGD 值变化的曲线。从图中可以看出, 随着投影维分段数  $k$  的增加, IGD 的值越好; 在投影维分段数  $k$  设置为 2 及以后, IGD 值的变化维持在了一个很小的范围之内; 划分投影格对种群进化起促进作用, 随着投影维划分段数的增加, 进化的投影格也就越多, 种群进化的时间成本也就越高, 从算法求解时间方面考虑, 建议将投影维分段数  $k$  设置为 2。

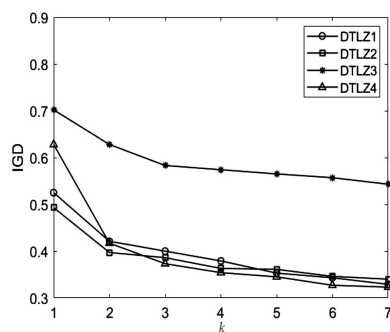


图 3 GPEA 在 DTLZ 测试问题上不同投影分段数的 IGD 变化曲线

图 4 表示在投影维数  $p$  和投影维分段数  $k$  不变的情况下, 自由维分段数  $g$  变化对 IGD 值变化的曲线。从图中可以看出, 随着自由维分段数  $g$  的增加, IGD 的值越好; 在自由维分段数  $g$  设置为 8 及以后, IGD 的值变化维持在了一个很小范围内; 自由维网格坐标划分对种群的进化起促进作用, 随着自由维划分段数  $g$  的增加, 单位网格坐标范围减小, 对应个体的网格排序值差异越大, 在网格适应度计算中网格排序占据主导地位。从算法的求解效率方面考虑, 建议将自由维分段数  $g$  设置为 8。

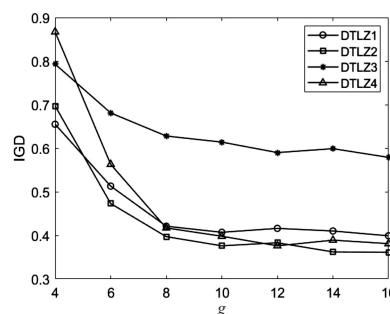


图 4 GPEA 在 DTLZ 测试问题上不同网格分段数的 IGD 变化曲线

表 1 给出了所有算法在 DTLZ1 ~ DTLZ4 测试问

表 1 不同算法在目标数不同的 DTLZ 测试问题上获得的 IGD 均值

Problem	M	GPEA	MOEA/D	GrEA	NSGA-III	NSGA/P	MOEA/DP
DTLZ1	3	0.029	0.032	0.058	0.043	0.031	0.026
	5	0.086	0.096	0.158	0.283	0.102	0.089
	7	0.293	0.329	0.573	0.957	0.406	0.318
	10	0.421	0.705	0.635	0.907	0.584	0.494
DTLZ2	3	0.035	0.039	0.051	0.038	0.026	0.027
	5	0.139	0.195	0.173	0.204	0.137	0.157
	7	0.318	0.321	0.345	0.369	0.341	0.297
	10	0.397	0.441	0.431	0.580	0.473	0.418
DTLZ3	3	0.193	0.255	0.306	0.392	0.216	0.229
	5	0.304	0.319	0.497	0.325	0.298	0.312
	7	0.438	0.585	0.619	0.703	0.487	0.501
	10	0.628	0.747	0.814	0.827	0.641	0.679

续表 1

Problem	M	GPEA	MOEA/D	GrEA	NSGA-III	NSGA/P	MOEA/DP
DTLZ4	3	0.027	0.039	0.078	0.047	0.041	0.031
	5	0.132	0.421	0.185	0.204	0.301	0.297
	7	0.340	0.603	0.369	0.315	0.437	0.453
	10	0.417	0.708	0.432	0.576	0.519	0.538

题上得到的 IGD 均值。从实验结果中可以看出: GPEA 在 DTLZ1 ~ DTLZ4 测试问题上表现良好,在超多目标问题空间中,种群的收敛性和多样性得到了很好的均衡,以下针对每个 DTLZ 测试问题详细分析算法的性能表现。

DTLZ1 和 DTLZ3 测试问题具有较多的局部帕累托前沿(Pareto Front, PF),为算法求解此类问题创造了更多的障碍。DTLZ2 和 DTLZ4 测试问题具有不同形状的 PF,为算法求解此类问题维持种群多样性提供了困难。从表 1 可以看出,GPEA 在大多数目标上取得了很好的结果,原因是投影维目标空间的投影格分

解策略可以协助种群跳出局部 PF,网格投影策略可以使种群收敛到 Pareto 前沿,网格适应度筛选策略可以使种群均匀的覆盖到 PF,提高了算法求解此类问题的鲁棒性。

图 5 为各算法在 7 目标 DTLZ4 测试问题上求得最终解集的平行坐标图,从图 5 可以看出 GPEA 在收敛性和分布性上取得了较好的结果;MOEA/D 存在某目标维解丢失的情况;GrEA 和 NSGA-III 在某目标维上存在局部解集;NSGA/P 和 MOEA/DP 的收敛性和分布性稍弱于 GPEA。

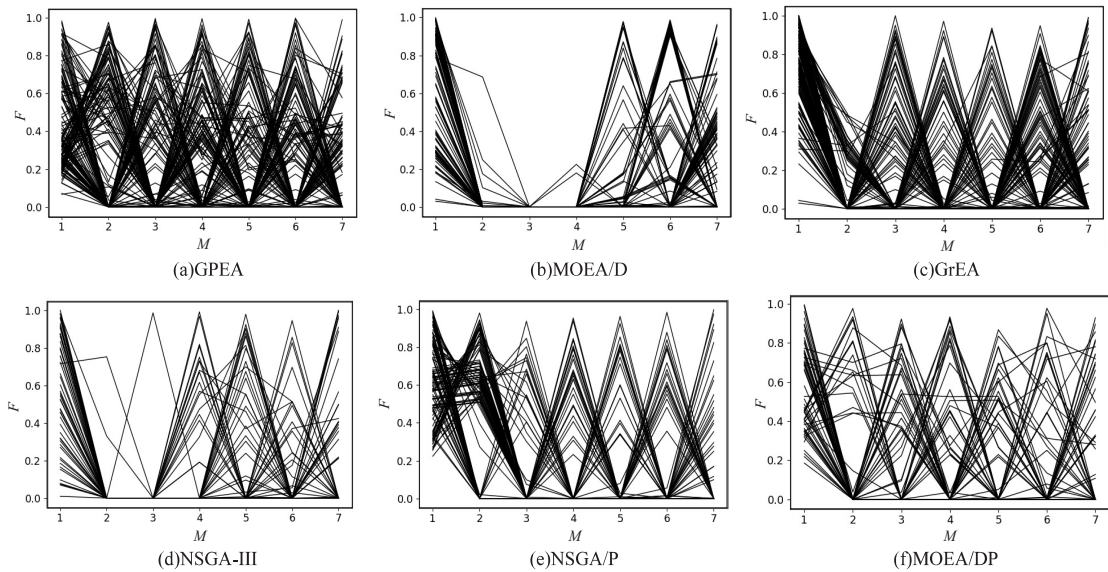


图 5 6 种算法在 7 目标 DTLZ4 测试问题上最终解集的平行坐标图

#### 4 结束语

针对超多目标优化问题使用多目标进化算法难以保证种群的收敛性和多样性的问题,提出了一种基于网格投影的超多目标进化算法。通过将目标空间拆分,分别构建投影维目标空间和自由维目标空间,使用投影格个体筛选策略和自由维目标空间个体筛选策略保持种群的收敛性和多样性,解决了 MOEA 求解超多目标优化问题难以平衡种群收敛性和多样性的问题。通过对标准测试函数设置不同参数实验,与 MOEA/D、GrEA、NSGA-III、MOEA/DP 和 NSGA/P 进行对比,实验结果表明,GPEA 能够很好地处理超多目标优化问题。下一步工作是改进投影格个体选择策略,研

究一种新的自由维目标空间适应度函数,并将算法与实际应用更好地结合起来。

#### 参考文献:

- [1] ZHOU A, QU B Y, LI H, et al. Multiobjective evolutionary algorithms: A survey of the state of the art [J]. Swarm and Evolutionary Computation, 2011, 1(1): 32-49.
- [2] LI K, WANG R, ZHANG T, et al. Evolutionary many-objective optimization: A comparative study of the state-of-the-art [J]. IEEE Access, 2018, 6: 26194-26214.
- [3] 万逸飞, 彭力. 基于协同多目标算法的多机器人路径规划 [J]. 信息与控制, 2020, 49(2): 139-146.
- [4] 韩锐, 吴军, 廖清芬, 等. 基于 NSGA-III 算法的光-水-火电机 AGC 协调优化策略 [J]. 智慧电力, 2022, 50

- (1):45–52.
- [5] 鲁宇明,史 册,黎 明,等. 基于改进 MOEA 算法的零件加工布局优化研究[J]. 机械设计,2021,38(5):49–56.
- [6] 王丽萍,任 宇,邱启仓,等. 多目标进化算法性能评价指标研究综述[J]. 计算机学报,2021,44(8):1590–1619.
- [7] 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.
- [8] ZHANG Q, LI H. MOEA/D: a multiobjective evolutionary algorithm based on decomposition[J]. IEEE Transactions on Evolutionary Computation, 2007, 11(6):712–731.
- [9] ZITZLER E, KÜNZLI S. Indicator-based selection in multiobjective search[C]//International conference on parallel problem solving from nature. Berlin: Springer, 2004.
- [10] CHEN H, TIAN Y, PEDRYCZ W, et al. Hyperplane assisted evolutionary algorithm for many-objective optimization problems[J]. IEEE Transactions on Cybernetics, 2019, 50(7):1–14.
- [11] DEB K, JAIN H. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints[J]. IEEE Transactions on Evolutionary Computation, 2013, 18(4):577–601.
- [12] TRAUTMANN H, WAGNER T, BROCKHOFF D. R2-EMOA: Focused multiobjective search using R2-indicator-based selection[C]//International conference on learning and intelligent optimization. Berlin: Springer, 2013:70–74.
- [13] 刘 宝. 基于投影面的多目标优化问题目标分解进化算法[D]. 沈阳: 沈阳化工大学, 2021.
- [14] DHIMAN G, KUMAR V. KnRVEA: A hybrid evolutionary algorithm based on knee points and reference vector adaptation strategies for many-objective optimization[J]. Applied Intelligence, 2019, 49(7):2434–2460.
- [15] 马畅畅,汪 坤,鹿晓梦,等. 一种确定目标域多目标优化算法 NSGA/P[J]. 计算机技术与发展, 2022, 32(5):15–21.
- [16] LU X, YANG S, PENG F, et al. An evolutionary algorithm for multi-objective optimization problem based on projection plane: MOEA/P[C]//5th international conference on algorithms, computing and systems. New York: Association for Computing Machinery, 2021:98–104.
- [17] JIANG S, YANG S. A strength Pareto evolutionary algorithm based on reference direction for multiobjective and many-objective optimization[J]. IEEE Transactions on Evolutionary Computation, 2017, 21(3):329–346.
- [18] YANG S, LI M, LIU X, et al. A grid-based evolutionary algorithm for many-objective optimization[J]. IEEE Transactions on Evolutionary Computation, 2013, 17(5):721–736.
- [19] DEB K, THIELE L, LAUMANN S M, et al. Scalable multi-objective optimization test problems[C]//Congress on evolutionary computation. Honolulu: IEEE, 2002:825–830.
- [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.