Journal of Computer Applications
计算机应用, 2021, 41(1): 22 - 28
ISSN 1001-9081
CODEN JYIIDU
2021-01-10
http://www.joca.cn

文章编号:1001-9081(2021)01-0022-07
DOI:10.11772/j.issn.1001-9081.2020060891

基于最小距离和聚合策略的分解多目标进化算法

李二超*,李康伟
(兰州理工大学 电气工程与信息工程学院,兰州 730050)
(*通信作者电子邮箱lecstarr@163.com)

摘　要:针对基于帕累托(Pareto)支配的多目标进化算法在解决高维问题时选择压力降低,以及基于分解的多目标进化算法在提高收敛性和分布性的同时降低了种群多样性的问题,提出了一种基于最小距离和聚合策略的分解多目标进化算法。首先,使用基于角度分解的技术将目标空间分解为指定个数的子空间来提高种群的多样性;然后,在生成新解的过程中加入基于聚合的交叉邻域方法,使生成的新解更接近于父代解;最后,分两阶段在每个子空间内基于最小距离和聚合策略来选择解以提高收敛性和分布性。为了验证所提算法的可行性,采用标准测试函数ZDT和DTLZ进行仿真实验,结果表明所提算法的总体性能均优于经典的基于分解的多目标进化算法(MOEA/D)、MOEA/D-DE、NSGA-Ⅲ和GrEA。可见,所提算法在提高多样性的同时可以有效平衡收敛性和多样性。

关键词:进化优化算法;多目标优化问题;收敛性;多样性;分布性;分解
中图分类号:TP181　　**文献标志码**:A

Decomposition based many-objective evolutionary algorithm based on minimum distance and aggregation strategy

LI Erchao*, LI Kangwei
(*College of Electrical Engineering and Information Engineering, Lanzhou University of Technology, Lanzhou Gansu* 730050, *China*)

Abstract: Concerning the issue that the selection pressure of Pareto control based many-objective evolutionary algorithm is reduced when solving the problem of high-dimension and the diversity of the population is reduced of many-objective evolutionary algorithm based on decomposition when improving convergence and distribution, a decomposition based many-objective evolutionary algorithm based on minimum distance and aggregation strategy was proposed. Firstly, the angle decomposition based technique was used to decompose the target space into a specified number of subspaces in order to improve the diversity of population. Then, the method of cross neighborhood based on aggregation was added in the process of generating new solution, making the generated new solution closer to the parent solution. Finally, the convergence and distribution were improved by selecting solutions in each subspace based on minimum distance and aggregation strategy in two stages. In order to verify the feasibility of the algorithm, benchmark functions ZDT and DTLZ were used to conduct simulation experiments. The results show that the performance of the proposed algorithm is superior to those of the classical MOEA/D (Multi-Objective Evolutionary Algorithm based on Decomposition), MOEA/D-DE (MOEA/D based on Differential Evolution), NSGA-Ⅲ (Nondominated Sorting Genetic Algorithms Ⅲ) and GrEA (Grid-based Evolutionary Algorithm). It can be seen that the proposed algorithm can effectively balance convergence and diversity while improving diversity.

Key words: evolutionary optimization algorithm; Many-objective Optimization Problem (MOP); convergence; diversity; distribution; decomposition

0　引言

进化算法是一种基于群体智能的启发式优化算法,它通过对生物进化的繁殖、变异、重组和选择等过程的模拟来达到优化的效果。这种算法对于求解多目标优化问题具有很好的效果,基于进化算法,研究者们相继提出了很多解决不同多目标优化问题(Many-objective Optimization Problem, MOP)[1]的优化算法。

以非支配排序遗传算法(Nondominated Sorting Genetic Algorithm, NSGA)[2]为代表的第一代进化算法,基于Pareto支配关系以及多样性保留策略对个体进行选择,该算法在选择解之前先将种群根据支配关系进行分层,以便能够有更大的机会选择出更好的个体遗传到下一代,并且在选择解的过程中加入了适应度共享策略,保持了解的多样性。但是,NSGA无法很好地保证高质量的解遗传到下一代,而使一些好的解丢失,并且需要指定共享参数使其依赖于共享的概念。因此,在非支配排序遗传算法(NSGA)、向量评估遗传算法(Vector Evaluated Genetic Algorithm, VEGA)[3]、小生境Pareto遗传算法(Niched Pareto Genetic Algorithm, NPGA)[4]等基础上引入精英保留策略提出了第二代进化算法,例如强度帕累托进化算法(Strength Pareto Evolutionary Algorithm, SPEA)[5]、改进的强度帕累托进化算法(Strength Pareto Evolutionary Algorithm

收稿日期:2020-05-30;**修回日期**:2020-07-28;**录用日期**:2020-08-03。　**基金项目**:国家自然科学基金资助项目(61763026)。
作者简介:李二超(1980—),男,河北保定人,教授,博士,主要研究方向:人工智能、多目标优化、机器人控制;　李康伟(1994—),男,甘肃天水人,硕士研究生,主要研究方向:多目标优化。

Two, SPEA2)[6]、基于非支配排序的多目标进化算法(Nondominated Sorting Genetic Algorithm Ⅱ, NSGA-Ⅱ)[7]等。第二代进化算法的代表算法NSGA-Ⅱ与NSGA类似,首先将整个种群根据Pareto非支配排序进行分层,其中前一层的解支配后一层的解,而同层的解之间互不支配;在选择解时使用精英解保留策略逐层选择解以保证收敛性,在最后一层采用拥挤距离的选择方法以保证解的多样性。Cai等[8]为了评估超多目标优化问题的最优前沿逼近真实前沿的程度,提出了一种基于参考向量的多样性指标(Diversity Indicator based on Reference vectors, DIR)。在DIR中,首先生成一组均匀分布的参考向量,然后根据每个解关联的参考向量数量计算解的覆盖率,最后由所有解的覆盖率的标准差决定解的多样性。DIR的值越小,多样性越好。将DIR与NSGA-Ⅱ结合提出了DIR增强的NSGA-Ⅱ(DIR-enhanced NSGA-Ⅱ, d-NSGA-Ⅱ)。这种第二代多目标进化算法得到的近似解不能均匀地分布在Pareto最优前沿上。为了提高多样性,Deb等[9]在NSGA-Ⅱ算法中引入参考点,将每个解关联在相应的参考向量上,提出了基于参考点的多目标遗传算法——NSGA-Ⅲ(Nondominated Sorting Genetic Algorithm Ⅲ),NSGA-Ⅲ为第三代进化算法的代表算法,它结合了非支配排序和基于参考点选择的优点,在保证收敛性的基础上提高了种群的分布性[10-11]。Bi等[12]在NSGA-Ⅲ的基础上加入了基于消元算子的新的选择机制,提出了基于消元算子的加强NSGA-Ⅲ多目标进化算法——NSGA-Ⅲ-EO(NSGA-Ⅲ based on Elimination Operator),此算法增加了解的选择压力,并且提高了种群的多样性,但是上述这类基于支配的多目标进化算法在解决高维问题时由于各目标之间互不支配[13-14],从而使选择压力降低。

基于参考点分解的多目标进化算法设置了一组参考向量,将目标空间分成若干个子区域来求解[15-16]。在算法当中每个子区域相当于一个子问题,通过对每个子种群同时进行优化来达到优化的效果。Zhang等[17]通过将一个多目标优化问题分解成一组单目标优化问题,提出了一种基于分解的多目标进化算法(Multi-Objective Evolutionary Algorithm based on Decomposition, MOEA/D)。MOEA/D算法在多目标优化问题尤其是高维多目标优化问题的收敛性和分布性等方面表现出了很好的效果,但是这种算法依然存在一些不足,比如在更新邻域子问题时会使其相邻子问题以外表现较好的解丢失,一个解同时出现在多个邻域当中而降低了解的多样性[18],为此近年来许多学者在此算法上做了更多的研究。Liu等[19]将均匀分布的参考向量分解成多个子空间,提出将一个多目标优化问题分解为若干个简单多目标子问题的多目标进化算法MOEA/D-M2M(MOEA/D based on decomposing MOP into a number of simple Multi-objective subproblems),其中,每个子空间对应一定大小的子种群,明显增加了解的多样性,而在解的选择过程中加入了Pareto非支配排序方法使其均匀性有所下降。Li等[20]将MOEA/D与NSGA-Ⅱ结合,并且加入了差分进化(Differential Evolution, DE)的策略,另外,为了改善MOEA/D的收敛性能,将局部搜索的思想融入MOEA/D算法中,提出了解决复杂Pareto解集的多目标进化算法——MOEA/D-DE(MOEA/D based on Differential Evolution),此算法使用聚合的方法选择解的过程虽然提高了解的均匀性但是多样性仍有不足。本文针对基于支配的多目标进化算法选择压力不高和均匀性差以及MOEA/D多样性不足的问题,将均匀分布的参考向量根据角度分解成多个子空间来增加多样性,并且引入分两阶段基于最小距离和聚合方法选择解的策略(Two-stage selection strategy based on minimum Distance and Aggregation method, TDA)来平衡收敛性、多样性以及均匀性,提出了基于最小距离和聚合策略的分解多目标进化算法——MOEA-TDA。

1　问题描述

1.1　多目标优化

多目标优化问题的各目标之间相互冲突,并且随着目标个数的增加问题的复杂度越高,且越不易求解[21-22]。多目标优化通常用于生产调度和路径规划等实际问题当中,通常情况下优化过程不能使每个目标同时达到最优,而只能找到一组折中解来表示最优解,因此多目标优化的目的是如何用特定的算法来找到一组最优解。一般情况下,将两目标和三目标的问题称为多目标优化问题,将三目标以上的问题称为超多目标优化问题[23]。

多目标优化算法有最大化目标函数值和最小化目标函数值两种[24],此处以最小化目标函数值为例:

$$\min\ \boldsymbol{F}(\boldsymbol{x})=(f_1(\boldsymbol{x}),f_2(\boldsymbol{x}),\cdots,f_M(\boldsymbol{x}))^{\mathrm{T}} \tag{1}$$

$$\text{s.t.}\ \ \boldsymbol{x}\in\boldsymbol{\Omega}$$

$$g_i(\boldsymbol{x})\leqslant 0;\quad i=1,2,\cdots,p$$

$$h_i(\boldsymbol{x})=0;\quad i=1,2,\cdots,q$$

其中:$\boldsymbol{\Omega}$代表决策空间,$\boldsymbol{x}=(x_1,x_2,\cdots,x_n)^{\mathrm{T}}\in\boldsymbol{\Omega}$为$n$维决策变量;$\boldsymbol{\Theta}$代表目标空间,$\boldsymbol{F}(\boldsymbol{x})\in\boldsymbol{\Theta}$为$M$个目标函数;$g_i(\boldsymbol{x})\leqslant 0(i=1,2,\cdots,p)$为$p$个不等式约束条件;$h_i(\boldsymbol{x})=0(i=1,2,\cdots,q)$为$q$个等式约束条件。以下是几个关于Pareto支配[25-26]的定义。

Pareto支配　对于任意两个解$\boldsymbol{x},\boldsymbol{y}\in\boldsymbol{\Omega}$,若$\forall i\in\{1,2,\cdots,M\}$有$f_i(\boldsymbol{x})\leqslant f_i(\boldsymbol{y})$,且$\exists j\in\{1,2,\cdots,M\}$满足$f_j(\boldsymbol{x})<f_j(\boldsymbol{y})$,则$\boldsymbol{x}$支配$\boldsymbol{y}$(记为$\boldsymbol{x}\prec\boldsymbol{y}$)。

Pareto最优解　如果$\boldsymbol{x}\in\boldsymbol{\Omega}$,不存在$\boldsymbol{y}\in\boldsymbol{\Omega}$使得$\boldsymbol{y}\prec\boldsymbol{x}$成立,则称$\boldsymbol{x}$是Pareto最优解。

Pareto最优解集　所有Pareto最优解组成的解集,即$P=\{\boldsymbol{x}|\boldsymbol{x}\in\boldsymbol{\Omega}\wedge\neg\exists\boldsymbol{y}\in\boldsymbol{\Omega},\boldsymbol{y}\prec\boldsymbol{x}\}$。

Pareto最优前沿　Pareto最优解对应的目标函数值所组成的解集称为Pareto最优前沿,即:$PF=\{\boldsymbol{F}(\boldsymbol{x})=(f_1(\boldsymbol{x}),f_2(\boldsymbol{x}),\cdots,f_M(\boldsymbol{x}))^{\mathrm{T}}|\boldsymbol{x}\in P\}$。

1.2　分解方法

在高维多目标优化算法中通常采用分解的方法将一个多目标优化问题分解为一组单目标问题进行求解,在本文中采用常用的基于惩罚的边界交叉法(Penalty-based Boundary Intersection, PBI)作为分解方法。第i个子问题定义如式(2)所示:

$$\begin{aligned}&\min\ g^{pbi}(\boldsymbol{x}|\boldsymbol{w}^i,\boldsymbol{z}^*)=d_1^i+\theta d_2^i;\\&d_1^i=\left(\boldsymbol{F}(\boldsymbol{x})-\boldsymbol{z}^*\right)^{\mathrm{T}}\boldsymbol{w}^i/\|\boldsymbol{w}^i\|,\\&d_2^i=\left\|\boldsymbol{F}(\boldsymbol{x})-\boldsymbol{z}^*-\left(d_1^i/\|\boldsymbol{w}^i\|\right)\boldsymbol{w}^i\right\|\\&\text{s.t.}\ \ \boldsymbol{x}\in\boldsymbol{\Omega}\end{aligned} \tag{2}$$

其中:θ为惩罚因子;$\|\cdot\|$为L_2范数;$\boldsymbol{w}^i=(w_1,w_2,\cdots,w_m)^{\mathrm{T}}$为分解后第$i$个子问题的权重向量。

1.3　权重向量的生成

在多目标优化算法中常用比较成熟的Das和Dennis系统方案[27]分解技术,在单位超平面上生成一组均匀分布的参考点,参考点的数目N和每维目标上的划分份数H的关系如式(3)所示:

$$N=C_{H+m-1}^{m-1} \tag{3}$$

其中m表示目标的维数,每一个参考向量$\boldsymbol{w}^i(i=1,2,\cdots,m)$中的所有元素之和为1,并且每一维元素都取值于$\{0/H,1/H,\cdots,H/H\}$。图1表示3目标问题参考点生成的示意图,$r_1$、$r_2$和$r_3$为目标的维数。

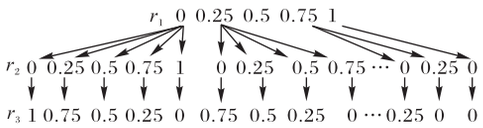


图1 权重向量的生成

Fig. 1 Generation of weight vectors

2 本文算法

2.1 基本思想和主要框架

基于分解的多目标进化算法对于解的分布性有很大的提升，但是在使用聚合方法选择解的过程中降低了解的多样性。本文在基于参考向量分解技术的基础上根据角度将种群分解成一组特定大小的子种群，然后分两阶段在每个子种群内使用最小距离和聚合的方法选择解，这种分解技术可以有效提高种群的多样性。为了提高进化效率，在生成新解的过程中引入邻域，由于邻域中的最优解较为相似，所以在交叉变异的过程中能够促进进化，并且能够降低计算复杂度和提高局部搜索能力。

MOEA-TDA主要分为5个步骤：1)初始化种群Pop、权重向量集W；2)求解交叉邻域A；3)生成新解；4)根据角度分解子种群；5)根据TDA策略选择解。MOEA-TDA总框架伪代码如算法1所示。

算法1 MOEA-TDA总框架。

输入：

1）种群最大迭代数$Maxgen$；

2）种群规模N；

3）目标个数M；

4）交叉邻域大小T；

5）权重向量集$W=\{\boldsymbol{w}^1,\boldsymbol{w}^2,\cdots,\boldsymbol{w}^N\}$。

输出：

最优解集：$Pop \leftarrow \bigcup_{i=1}^{N} subpop(i)$。

1）初始化种群Pop；
2）while 不满足终止条件 do
3） 基于PBI聚合关联方法选择交叉邻域A；
4） for $i \leftarrow 1$ to N do
5） 从交叉邻域A中选择父代解并根据差分进化和多项式变异算子生成子代解；
6） 更新理想点$\boldsymbol{z}$；
7） end for
8） 将父代解和子代解混合生成$Mixpop$；
9） 根据角度分解目标空间以生成子种群 $\bigcup_{i=1}^{N} subpop(i) \leftarrow Mixpop$；
10） 根据TDA策略选择解；
11） 输出最优解集Pop；
12）end

2.2 基于聚合的交叉邻域

在MOEA/D中采用均匀分布的权重向量生成了不变的邻域，使得每个子问题都含有很多邻近的子问题，邻域内的子问题具有相似的最优解，所以对子问题的进化有一定的帮助。为了进一步提高算法的进化效率和局部搜索能力，提出基于聚合的邻域选择策略。在MOEA/D等经典的多目标优化算法中每个子问题的邻域都是根据相邻的权重向量所确定的，这样使得进化前期每个权重向量所对应的子问题都不是离其最近的，所以在差分进化的过程当中局部搜索能力不高，本算法中提出的基于聚合的交叉邻域策略不再是将离权重向量的欧氏距离最近的个体作为邻域解，而是计算出权重向量与每个个体之间的PBI值，然后选出PBI值最小的T个解为邻域，这时邻域随着每一代的进化而发生自适应变化，有利于提高种群的局部搜索能力。伪代码如算法2所示。

算法2 基于聚合的交叉邻域。

输入：

1）父代种群Pop；

2）权重向量集$W=\{\boldsymbol{w}^1,\boldsymbol{w}^2,\cdots,\boldsymbol{w}^N\}$。

输出：

交叉邻域索引A。

1）for $\boldsymbol{w} \in W$ do
2） for $\boldsymbol{x} \in Pop$ do
3） 计算目标函数值和权重向量之间的cos值
4） end for
5） 找出离每个权重向量PBI值最小的T个个体索引值
6）end for
7）输出索引集A

在进化过程中对于每个权重向量除了原本与其相关联的最优解外，再找出与权重向量的PBI值最小的两个解来进行差分进化。本文使用MOEA/D-DE中提出的差分进化和多项式变异来产生子代个体，差分进化中有交叉、变异和选择几个主要的步骤，这种进化算法比较适合解决连续的优化问题，它能引导新解产生的方向。

2.3 基于角度的目标空间分解策略

首先在目标空间中产生N个均匀分布的参考向量，再根据参考向量将目标空间划分为指定数量的子区间，每一个子区间对应唯一的子种群，其中每一代的种群采用基于角度的划分方式将目标空间划分为N个子区间$\{\Omega^1,\Omega^2,\cdots,\Omega^N\}$以提高所求最优解集的多样性，对于每一个子区间：

$$\Omega^i=\left\{f'(\boldsymbol{x})\in \mathbf{R}_+^m \middle| \left\langle f'(\boldsymbol{x}),\boldsymbol{w}^i\right\rangle \leqslant \left\langle f'(\boldsymbol{x}),\boldsymbol{w}^j\right\rangle\right\} \tag{4}$$

其中：$f'(\boldsymbol{x})=f(\boldsymbol{x})-\boldsymbol{Z}^*$，$i,j\in\{1,2,\cdots,N\}$，$\boldsymbol{Z}^*=(f_1^{\min},f_2^{\min},\cdots,f_m^{\min})$，$\boldsymbol{w}$为预先设定的参考向量。在整个种群当中，$\left\langle f'(\boldsymbol{x}),\boldsymbol{w}^i\right\rangle$为$f'(\boldsymbol{x})$与$\boldsymbol{w}^i$之间的夹角，$\Omega^i$表示第$i$个向量$\boldsymbol{w}^i$与其夹角最小的解所组成的子种群，这里的夹角用余弦值来计算，以保证和夹角的变化保持一致。计算公式如式(5)所示：

$$\cos\theta=\frac{\boldsymbol{w}\cdot f'(\boldsymbol{x})}{\left\|f'(\boldsymbol{x})\right\|} \tag{5}$$

基于角度的目标空间分解策略原理如图2所示，如图选出任意两个权重向量$\boldsymbol{w}^i$和$\boldsymbol{w}^j$，在其周围分布着6个不同的个体。在传统的多目标优化算法当中一般使用基于欧氏距离的计算方法找出与权重向量相关联的个体，例如，在分解过程中根据欧氏距离只会将$\boldsymbol{x}_2$和$\boldsymbol{x}_4$划分到同一区间内，这种分解方法使算法的收敛性有所提高，但是在维持多样性方面有一定的缺陷。而使用基于角度的目标空间的分解策略划分子空间时，在同一子区间内不仅包含$\boldsymbol{x}_2$和$\boldsymbol{x}_4$，而且也包含$\boldsymbol{x}_6$，这样在选择解的过程中会很大程度上维持种群的多样性。

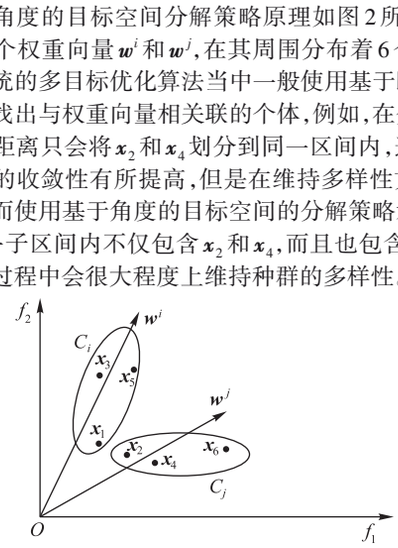


图2 基于角度的目标空间的分解策略

Fig. 2 Decomposition strategy of target space based on angle

2.4 基于最小距离和聚合策略选择解

根据角度分解目标空间之后会使每个权重向量关联到一

个子区间,因此可以在每个子区间内分两阶段基于最小距离和聚合策略单独选择解。由于在分解完子区间后有一些区间内关联不到个体,所以根据此策略选择解时,需要分以下两种情况进行:

1)子区间内有关联个体。

在这种情况下,当进化代数达到临界代数之前基于最小距离选择解,以提高算法的收敛性;当进化代数达到临界代数之后基于PBI聚合的策略选择解,以提高解的分布性。在本算法中基于距离和基于角度选择解的临界代数是根据多次实验结果测试出来的。这里的距离由式(6)给出,公式当中的各参数与式(2)中相同。

$$d = \left(\boldsymbol{F}(\boldsymbol{x}) - \boldsymbol{z}^*\right)^{\mathrm{T}} \boldsymbol{w}^i \Big/ \left\| \boldsymbol{w}^i \right\| \tag{6}$$

2)子区间内无关联个体。

当子区间内无关联个体时,不能直接选择解,此时需要计算子代和父代组成的混合种群在该区间对应的权重向量下的PBI值,然后选择出混合种群中最小的PBI值所对应的个体作为该区间的最优解。两种情况下的算法流程如图3所示。

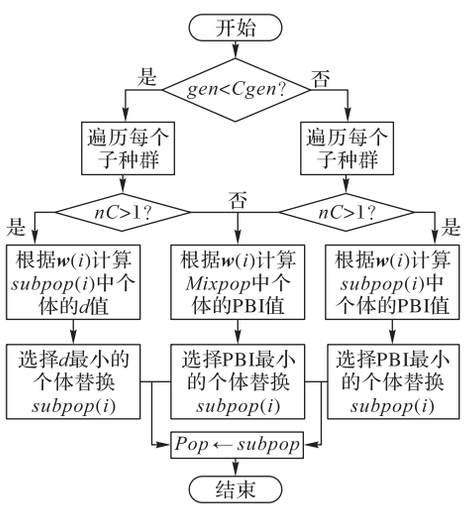


图3　基于最小距离和聚合策略的解选择

Fig. 3　Solution selection based on minimum distance and aggregation strategy

图3中:*gen*为当前运行代数,*Cgen*为临界代数,*nC*为第*i*个子种群中的个体数,*subpop*(*i*)为第*i*个子种群,$\boldsymbol{w}(i)$为第*i*个子种群相关联的权重向量,*Mixpop*为子代和父代的混合种群,*Pop*为最终输出种群。

3　实验和结果分析

为了测试MOEA-TDA的性能,分别选取了ZDT[28]和DTLZ[29]系列测试问题。其中:ZDT系列测试问题主要选取具有代表性的ZDT1~ZDT4和ZDT6;而DTLZ系列测试问题选取DTLZ1~DTLZ5,主要测试算法在高维情况下的性能。

ZDT1和ZDT4为连续凸型测试函数,但前者的偏约束个数大于后者;ZDT2为连续凹型测试函数;而ZDT3为前沿断续的测试函数;以上均为基因型均匀的测试函数。ZDT6为连续凸型但基因型非均匀的测试函数。对于DTLZ1,目标函数值位于一个线性超平面上,并且难以收敛到最优解;DTLZ2的Pareto最优解集位于单位球的第一象限,它可用于探测一个MOEA对高维目标的进化搜索能力;DTLZ3不仅包含局部Pareto最优解集还包含一个全局Pareto最优解集,可测试全局收敛能力;DTLZ4测试MOEA维持一个良好的分布解集的能力[30];DTLZ5的Pareto最优解集聚集于一条曲线上,用于测试MOEA收敛到一条曲线的能力。本文主要研究收敛性和多样性问题,因此使用以上测试函数进行实验。

为了验证可行性,本算法与4种经典的算法MOEA/D、MOEA/D-DE、NSGA-Ⅲ和基于网格的多目标进化算法GrEA[31]进行了比较。

3.1　参数设置

为了公平比较,所有算法的种群规模和运行代数均保持一致,对于GrEA算法,由于自身的机制限制,将种群规模设置为4的倍数。各测试问题的参数设置如表1所示。

表1　DTLZ测试问题的参数设置

Tab. 1　Parameter setting of DTLZ test problem

函数	参数K	目标维数M	决策变量维数D
ZDT2-6		2	10
DTLZ1-5		3	10
DTLZ1	5	5,6,8,10	$M+K-1$
DTLZ2-5	10	5,6,8,10	$M+K-1$

对于3、6、8以及10维问题,所有算法的种群规模分别设置为300、252、120以及220;而对于5维问题,GrEA算法的种群规模设置为212,其他算法设置为210,GrEA算法的*div*设为10。ZDT系列测试问题的迭代次数设为100,DTLZ系列测试问题的迭代次数设为1 000,交叉指数*mu*=30,变异指数*mum*=20,变异概率$P_m = 1/D$,交叉概率$CR = 1.0$,惩罚因子$\theta = 5$,差分进化参数F=0.5,所有算法在每个测试函数上独立运行20次。

3.2　性能指标

为了能够更精确地比较算法的性能优劣,通常需要做定量的对比,在本文中使用较为广泛的综合性能指标反世代距离(Inverted Generational Distance, IGD)[32-33]和超体积(HyperVolume, HV)[34]作为评估指标。

反世代距离(IGD)兼顾了近似解集的收敛性和多样性,其通过真实前沿上均匀采样的一组参考点与求得的近似最优解集之间的平均距离来计算。P^*为Pareto真实前沿上均匀采样得到的一组参考点,P为使用优化算法求出的一组近似最优解集。则IGD指标的计算如式(7)所示:

$$IGD(P, P^*) = \frac{1}{|P^*|} \sum_{\boldsymbol{v} \in P^*} dist(\boldsymbol{v}, P) \tag{7}$$

其中:$|P^*|$表示P^*的基数,即解集P^*中解的个数;$dist(\boldsymbol{v}, P)$表示$\boldsymbol{v} \in P^*$到其最近解的欧氏距离。若IGD值越小,则所求得的近似最优解集的性能越好,即更接近于真实前沿PF。

另一个指标超体积HV也用来同时评估所得近似最优解集的收敛性和多样性,其不需要真实Pareto真实前沿而通过在目标空间设定一个参考点来求得。假设$\boldsymbol{u}^r = (u_1^r, u_2^r, \cdots, u_m^r)^{\mathrm{T}}$为目标空间中被所有的Pareto最优解支配的一个参考点,则超体积表示目标空间中以$\boldsymbol{u}^r$为边界被最优解集P中的解所支配的区域的大小,HV指标的计算如式(8)所示:

$$HV(P) = VOL\Big(\bigcup_{\boldsymbol{x} \in P} [f_1(\boldsymbol{x}), u_1^r] \times [f_2(\boldsymbol{x}), u_2^r] \times \cdots \times [f_m(\boldsymbol{x}), u_m^r]\Big) \tag{8}$$

其中,$VOL(\cdot)$表示勒贝格计量。HV越大说明最优前沿P的性能越好。

3.3　实验结果分析

为了更客观地显示出所提算法的优越性,分别对所有算法的IGD和HV指标独立计算20次,所有算法在ZDT测试问题下的IGD和HV指标的均值和标准差如表2所示,在DTLZ

测试问题下的IGD和HV指标均值和标准差分别如表3~4所示，并且在表中用粗体标出了最好的值。

表2 5种算法在ZDT测试集上的平均IGD和HV均值以及标准差

Tab. 2 Average IGD, average HV mean and standard deviation of five algorithms on DTLZ test set

函数	性能指标	MOEA-TDA		MOEA/D		MOEA/D-DE		NSGA-Ⅲ		GrEA	
		均值	标准差	均值	标准差	均值	标准差	均值	标准差	均值	标准差
ZDT1	IGD	**2.955E-3**	**7.71E-5**	5.408E-3	3.68E-4	2.657E-2	1.33E-2	5.106E-2	2.96E-2	6.231E-2	1.10E-2
	HV	**8.721E-1**	**6.62E-4**	8.666E-1	1.05E-3	8.359E-1	1.12E-2	8.278E-1	1.97E-2	7.758E-1	2.14E-2
ZDT2	IGD	**3.172E-3**	**9.99E-4**	3.685E-3	1.03E-3	2.955E-2	5.98E-3	8.988E-3	3.76E-3	6.545E-3	3.75E-4
	HV	**5.386E-1**	**2.58E-3**	5.372E-1	2.50E-3	4.888E-1	1.08E-2	5.246E-1	9.69E-3	5.353E-1	3.91E-4
ZDT3	IGD	**1.261E-2**	**9.67E-3**	5.781E-2	4.81E-2	1.129E-1	2.12E-2	1.434E-2	1.23E-2	1.638E-2	1.11E-2
	HV	1.006E+0	1.43E-2	9.467E-1	5.80E-2	8.638E-1	1.56E-2	1.001E+0	1.15E-2	**1.014E+0**	**1.18E-2**
ZDT4	IGD	**3.048E-3**	**5.06E-4**	1.438E+0	7.05E-1	3.733E-1	1.23E-1	3.985E-1	1.23E-1	6.488E-2	2.93E-2
	HV	**8.721E-1**	**1.21E-3**	3.474E-2	7.14E-2	3.926E-1	1.32E-1	3.439E-1	1.24E-1	7.735E-1	4.56E-2
ZDT6	IGD	**2.289E-3**	**2.63E-5**	4.005E-3	1.15E-3	2.808E-2	3.25E-2	2.526E-1	1.32E-1	7.777E-2	1.37E-2
	HV	**4.338E-1**	**1.02E-5**	4.318E-1	2.08E-3	4.081E-1	3.09E-2	1.426E-1	8.81E-2	3.219E-1	1.94E-2

表3 5种算法在DTLZ测试集上的平均IGD均值及标准差

Tab. 3 Average IGD mean and standard deviation of five algorithms on DTLZ test set

函数	M	MOEA-TDA		MOEA/D		MOEA/D-DE		NSGA-Ⅲ		GrEA	
		均值	标准差	均值	标准差	均值	标准差	均值	标准差	均值	标准差
DTLZ1	3	**1.073E-2**	**1.69E-5**	2.678E-2	3.24E-5	1.267E-2	9.89E-6	3.135E-2	3.46E-3	2.125E-2	3.45E-3
	5	**5.046E-2**	**6.53E-4**	6.198E-2	2.39E-4	5.245E-2	4.26E-4	1.144E-1	3.84E-3	1.370E-1	3.65E-2
	6	**6.593E-2**	**8.02E-4**	8.192E-2	9.25E-4	6.594E-2	7.98E-4	3.431E-1	6.27E-2	7.747E-2	4.28E-2
	8	2.756E-1	1.23E-3	1.412E-1	6.35E-4	**1.089E-1**	**6.23E-4**	2.906E-1	3.57E-2	3.125E-1	6.43E-2
	10	1.151E-1	3.21E-3	2.837E-1	6.79E-4	**1.075E-1**	**1.36E-3**	1.533E-1	9.34E-4	3.772E+0	3.51E-2
DTLZ2	3	**2.815E-2**	**8.37E-5**	3.288E-2	3.76E-4	3.076E-2	7.18E-6	3.479E-2	1.24E-4	5.277E-2	1.29E-3
	5	**1.565E-1**	**1.03E-4**	1.742E-1	7.92E-4	1.987E-1	3.57E-4	1.924E-1	3.87E-4	1.726E-1	3.26E-3
	6	**2.105E-1**	**7.94E-5**	2.250E-1	9.17E-4	2.634E-1	6.86E-4	2.591E-1	6.51E-4	2.192E-1	5.37E-3
	8	**3.743E-1**	**8.52E-5**	3.824E-1	7.03E-4	3.974E-1	7.35E-4	8.026E-1	1.67E-2	3.696E-1	2.71E-3
	10	**3.595E-1**	**1.25E-4**	3.631E-1	1.09E-3	3.831E-1	9.23E-4	9.237E-1	1.06E-1	4.241E-1	9.98E-2
DTLZ3	3	**3.152E-2**	**3.57E-4**	3.783E-2	3.51E-4	3.763E-2	3.05E-3	4.917E-2	2.59E-3	6.903E-2	1.67E-2
	5	**2.286E-1**	**8.81E-4**	2.981E-1	6.18E-4	4.149E-1	1.29E-2	3.562E-1	7.15E-3	9.372E-1	3.82E-1
	6	**1.657E+0**	**8.36E-4**	4.281E+0	9.16E-4	4.984E+0	6.94E-1	2.367E+0	8.43E-3	7.718E-1	1.36E-1
	8	**3.456E+0**	**1.03E-3**	3.882E+0	1.00E-3	3.698E+0	3.54E-1	4.397E+0	2.94E-2	1.220E+0	6.28E-1
	10	5.795E+0	9.39E-2	7.929E+0	2.37E-3	**4.359E-1**	**2.94E-2**	4.444E+0	6.58E-2	1.689E+0	6.81E-1
DTLZ4	3	**2.812E-2**	**1.28E-5**	3.268E-2	2.18E-5	3.074E-2	1.66E-5	3.762E-2	1.53E-4	7.070E-2	2.35E-3
	5	**1.586E-1**	**5.39E-5**	1.729E-1	9.56E-5	1.731E-1	1.00E-4	1.749E-1	6.83E-4	1.645E-1	1.05E-2
	6	**2.112E-1**	**9.07E-5**	2.313E-1	2.57E-4	2.265E-1	3.12E-4	2.295E-1	8.39E-4	2.178E-1	3.75E-3
	8	3.787E-1	2.25E-4	3.869E-1	4.23E-4	**3.786E-1**	**3.77E-4**	4.206E-1	3.19E-3	3.798E-1	1.67E-2
	10	3.662E-1	3.77E-4	**3.562E-1**	**4.52E-4**	3.755E-1	3.28E-4	6.076E-1	7.29E-3	4.779E-1	6.35E-2
DTLZ5	3	**1.744E-2**	**1.69E-7**	1.766E-2	3.56E-7	1.756E-2	3.95E-5	4.834E-3	4.43E-4	2.153E-2	3.17E-4
	5	2.253E-2	6.27E-6	**2.246E-2**	**3.92E-6**	2.253E-2	8.83E-6	2.925E-2	9.16E-3	1.226E-1	5.21E-2
	6	2.733E-2	9.33E-6	2.708E-2	4.39E-6	**2.688E-2**	**2.79E-5**	3.615E-2	4.83E-2	1.954E-1	9.26E-2
	8	**6.839E-2**	**1.29E-5**	6.861E-2	3.38E-5	6.856E-2	2.87E-5	4.733E-1	6.13E-2	3.524E-1	7.39E-2
	10	**6.608E-2**	**2.91E-5**	6.609E-2	6.81E-4	6.609E-2	5.39E-4	3.319E-1	2.48E-1	2.617E-1	9.11E-2

对于二维多目标优化问题，由表2中的反世代距离(IGD)和超体积(HV)可以看出，MOEA-TDA与其他4种经典算法相比在收敛性和多样性方面有很大的提升，尤其对于ZDT4测试问题，其效果较为显著，但是对于ZDT1测试问题效果没有ZDT4效果明显，这说明MOEA-TDA对于解决偏约束个数较小的优化问题性能有很大的提升，这是由于第一阶段基于最小距离的选择策略能够有效提高解集的收敛能力。对于ZDT2，所提算法的效果优于其他算法，因此该算法对连续凹型测试问题也同样适用。对于非均匀的ZDT6，该算法也均优于其他对比算法，说明该算法能够有效地解决非均匀系列的测试函数。但是对于ZDT3这类不连续的测试问题而言，尽管该算法能够提高收敛性和多样性，但是它不能完全收敛到真实前沿上。因为在该算法中生成的权重向量都是均匀分布的，在利用角度分解目标空间的过程中会使前沿断续地方的权重向量也会参与到分解目标空间的过程中，从而使这些权重向量也匹配到解，因此对于前沿断续的问题效果并不好。从标准差可以看出，算法的鲁棒性也有所提高，所以算法MOEA-TDA在ZDT测试集上明显优于其他4类经典算法。

对于DTLZ系列测试函数，使用性能指标反世代距离IGD和超体积HV表示其结果如表3和表4。同样，由表3可以看出，算法MOEA-TDA在高维问题上也有较大的优势，对于DTLZ1测试问题而言，虽然收敛性提升较小，而且相对于NSGA-Ⅲ和GrEA最优解集分布更加均匀，但是MOEA/D-DE对DTLZ1的8、10目标效果比MOEA-TDA好。说明该算法对难以收敛到最优解集的测试问题性能有一定的提升，但是此策略对于DTLZ1问题随着目标维数的增加效果逐渐减弱。由表3~4可以看出，对于DTLZ2，不管是高维问题还是低维问题MOEA-TDA的优化效果均优于其他算法，说明基于最小距

离和聚合策略的选择方法对于高维目标的进化搜索能力有较大的提升。对于同时包含局部Pareto最优解集和全局Pareto最优解集的DTLZ3测试函数,从性能指标可以看出,维数较低时优化效果明显,但是维数增加到10维时优化效果显著降低,而MOEA/D-DE效果较好,说明当维数较低时MOEA-TDA能有效提高全局搜索能力,维数较高时MOEA/D-DE的全局搜索能力较强。DTLZ4主要测试解集的分布能力,在此算法中基于角度的目标空间分解策略主要用于提高最优解集的分布性,从实验结果看出该算法在目标数较低时效果较好,此外GrEA算法也能够很好地保持解集的分布性,因为在GrEA算法中使用到了基于网格的策略,对分布性也有较大的提升。而对于DTLZ5而言实验MOEA-TDA结果较差,这是由于使用了基于角度分解目标空间的策略,使其在高维空间中很难保证前沿退化问题的最优解集聚集到真实前沿上。

表4　5种算法在DTLZ测试集上的平均HV及标准差

Tab. 4　Average HV and standard deviation of five algorithms on DTLZ test set

函数	M	MOEA-TDA		MOEA/D		MOEA/D-DE		NSGA-Ⅲ		GrEA	
		均值	标准差	均值	标准差	均值	标准差	均值	标准差	均值	标准差
DTLZ1	3	**1.425E-1**	**3.84E-7**	1.287E-1	6.89E-6	1.413E-1	7.29E-6	1.342E-1	1.51E-3	1.276E-1	3.59E-4
	5	**4.926E-2**	**4.83E-6**	5.647E-3	4.95E-4	4.899E-2	3.87E-4	3.953E-2	9.27E-4	4.394E-2	7.13E-2
	6	**2.740E-2**	**9.37E-6**	1.743E-3	7.38E-4	2.732E-2	5.23E-4	1.073E-2	2.59E-3	2.694E-2	5.29E-2
	8	6.909E-3	2.49E-5	**8.295E-3**	**9.58E-4**	8.226E-3	8.26E-4	5.100E-3	1.05E-2	5.323E-3	2.73E-2
	10	**2.531E-3**	**9.54E-4**	2.181E-3	3.46E-4	2.517E-3	7.24E-3	1.008E-3	3.49E-2	0.000E+0	0.00E+0
DTLZ2	3	**7.724E-1**	**3.83E-6**	7.507E-1	1.06E-5	7.557E-1	2.49E-4	7.628E-1	3.57E-5	7.542E-1	3.89E-4
	5	**1.299E+0**	**1.08E-5**	1.214E+0	7.95E-4	1.121E+0	1.64E-3	1.092E+0	9.63E-5	1.303E+0	5.27E-4
	6	1.540E+0	1.04E-4	1.478E+0	2.10E-3	1.316E+0	8.31E-4	1.223E+0	3.62E-4	**1.542E+0**	**1.02E-3**
	8	**1.930E+0**	**3.49E-4**	1.902E+0	3.85E-3	1.575E+0	2.13E-3	1.376E+0	5.54E-2	1.924E+0	6.68E-3
	10	2.476E+0	7.94E-4	2.472E+0	3.44E-3	2.141E+0	6.39E-3	1.854E+0	2.73E-1	**2.511E+0**	**3.24E-3**
DTLZ3	3	**7.730E-1**	**5.374E-4**	7.441E-1	8.94E-4	7.353E-1	2.86E-3	7.083E-1	9.30E-4	7.383E-1	1.76E-2
	5	**1.302E+0**	**3.23E-3**	1.113E-1	6.384E-3	1.023E+0	7.49E-2	1.296E+0	5.94E-3	2.220E-1	8.41E-2
	6	**2.155E+0**	**8.94E-3**	1.116E+0	3.21E-1	1.965E+0	6.37E+0	1.238E-1	2.19E-3	5.440E-1	2.51E-1
	8	**1.334E+0**	**1.11E-2**	9.348E-1	1.57E+0	0.000E+0	0.00E+0	2.507E+0	3.33E-2	1.942E-1	4.31E-1
	10	0.000E+0	0.00E+0	0.000E+0	0.00E+0	1.808E+0	3.81E-3	**2.114E+0**	**8.39E-4**	1.948E-1	3.62E-1
DTLZ4	3	**7.730E-1**	**2.59E-4**	7.510E-1	3.43E-4	7.580E-1	4.36E-4	7.602E-1	9.37E-6	7.296E-1	3.62E-3
	5	1.286E+0	6.94E-4	1.246E+0	9.64E-4	1.241E+0	1.53E-3	1.236E+0	6.33E-5	**1.304E+0**	**7.62E-3**
	6	1.533E+0	1.03E-3	1.474E+0	3.28E-2	1.495E+0	9.15E-4	1.465E+0	2.58E-4	**1.541E+0**	**5.19E-2**
	8	1.937E+0	9.73E-4	**1.944E+0**	**1.12E-3**	1.916E+0	6.35E-3	1.518E+0	4.62E-4	1.863E+0	3.76E-2
	10	2.483E+0	4.85E-3	**2.499E+0**	**3.84E-2**	2.461E+0	1.96E-2	2.481E+0	4.11E-4	2.195E+0	8.96E-3
DTLZ5	3	**1.274E-1**	**7.63E-6**	1.273E-1	2.79E-6	1.273E-1	3.56E-6	1.326E-1	8.59E-3	1.252E-1	3.94E-4
	5	**9.095E-3**	**9.29E-5**	9.041E-3	8.24E-6	9.066E-3	6.59E-6	8.165E-3	5.34E-3	7.319E-3	1.37E-2
	6	**1.615E-3**	**1.08E-4**	1.613E-3	1.39E-4	1.609E-3	7.22E-4	1.153E-3	1.86E-2	1.253E-3	8.61E-2
	8	**1.929E-5**	**6.51E-6**	1.927E-5	4.29E-4	1.924E-5	2.56E-5	1.873E-5	4.38E-2	1.595E-5	6.43E-2
	10	**6.198E-8**	**3.25E-5**	6.194E-8	3.24E-4	6.165E-8	3.11E-5	2.147E-8	7.94E-1	8.948E-9	4.63E-2

NSGA-Ⅲ为非常经典的多目标优化算法,它结合了非支配排序和基于参考点选择的优点,在保证收敛性的基础上提高了种群的多样性,MOEA/D和MOEA/D-DE通常使用加权和法、Tchebycheff法和PBI的聚合方法,PBI的聚合方法比Tchebycheff法得到的最优解集更加均匀,主要用来求解非凸优化问题,在本文中为了与提出的算法保持一致采用PBI的聚合方法。从实验结果来看,与其他几个对比算法相比,MOEA/D-DE的结果较优,这是由于加入差分进化的原因。而基于网格的多目标优化算法GrEA在收敛性和多样性上表现也较为突出,不仅更加适合处理复杂的优化问题,而且对Pareto前沿的形状有很强的鲁棒性。MOEA-TDA在基于权重向量分解的基础上加入了基于角度的分解策略,进一步提高了种群的多样性并且保证了分布性;为了提高收敛性,在算法的前期使用基于最小距离的选择策略。综上,根据实验结果得出,提出的MOEA-TDA不仅能够提高收敛性和多样性,并且能够提高维持一个良好分布解集的能力。

4　结语

本文提出了一种分两阶段基于最小距离和聚合策略选择解的多目标进化算法——MOEA-TDA。首先,使用根据角度分解子空间的策略,有效地提高了解的多样性;其次,使用基于聚合的交叉邻域,提高了局部搜索能力;最后,分两阶段在每个子种群中分别根据最小距离和PBI聚合策略选择解,提高了解的收敛性。从实验结果可以得出,所提出的MOEA-TDA与其他4种经典的多目标优化算法相比在本文中的测试问题上综合性能都有明显的提高。

参考文献(References)

[1] COLLETTE Y, SIARRY P. Multiobjective Optimization: Principles and Case Studies [M]. Berlin: Springer, 2004: 273-316.

[2] SRINIVAS N, DEB K. Multiobjective optimization using non-dominated sorting in genetic algorithms [J]. Evolutionary Computation, 1994, 2(3): 221-248.

[3] SCHAFFER J D. Multiple objective optimization with vector evaluated genetic algorithms [C]// Proceedings of the 1st International Conference on Genetic Algorithms. Mahwah, NJ: Lawrence Erlbaum Associates, Inc., 1985: 93-100.

[4] HORN J, NAFPLIOTIS N, GOLDBERG D E. A niched Pareto genetic algorithm for multiobjective optimization [C]// Proceedings of the 1st IEEE Conference on Evolutionary Computation/1994 IEEE World Congress on Computational Intelligence. Piscataway: IEEE, 1994: 82-87.

[5] 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.

[6] ZITZLER E, LAUMANNS M, THIELE L. SPEA2: improving the strength Pareto evolutionary algorithm [C]// Proceedings of EUROGEN 2011 – Evolutionary and Deterministic Methods for Design, Optimization and Control with Applications to Industrial and Societal Problems. Athens: [s. n.], 2001: 95-100.

[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] CAI X, SUN H, FAN Z. A diversity indicator based on reference vectors for many-objective optimization [J]. Information Sciences, 2018, 430/431:467-486.

[9] 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, 2014, 18(4): 577-601.

[10] 袁源. 基于分解的多目标进化算法及其应用[D]. 北京:清华大学, 2015: 30-61. (YUAN Y. Decomposition-based multi-objective evolutionary algorithms and their applications [D]. Beijing: Tsinghua University, 2015: 30-61.)

[11] 巩敦卫,刘益萍,孙晓燕,等. 基于目标分解的高维多目标并行进化优化方法[J]. 自动化学报, 2015, 41(8): 1438-1451. (GONG D W, LIU Y P, SUN X Y, et al. Parallel many-objective evolutionary optimization using objectives decomposition [J]. Acta Automatica Sinica, 2015, 41(8): 1438-1451.)

[12] BI X, WANG C. An improved NSGA-Ⅲ algorithm based on elimination operator for many-objective optimization [J]. Memetic Computing, 2017, 9(4): 361-383.

[13] BANDYOPADHYAY S, MUKHERJEE A. An algorithm for many-objective optimization with reduced objective computations: a study in differential evolution [J]. IEEE Transactions on Evolutionary Computation, 2015, 19(3): 400-413.

[14] 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.

[15] ASAFUDDOULA M, RAY T, SARKER R A. A decomposition-based evolutionary algorithm for many objective optimization [J]. IEEE Transactions on Evolutionary Computation, 2015, 19(3): 445-460.

[16] LI Z, LIN K, NOUIOUA M, et al. DCDG-EA: dynamic convergence-diversity guided evolutionary algorithm for many-objective optimization [J]. Expert Systems with Applications, 2019, 118: 35-51.

[17] ZHANG Q, LI H. MOEA/D: a multiobjective evolutionary algorithm based on decomposition [J]. IEEE Transactions on Evolutionary Computation, 2007, 11(6): 712-731.

[18] HE Z, YEN G G. Many-objective evolutionary algorithm: objective space reduction and diversity improvement [J]. IEEE Transactions on Evolutionary Computation, 2016, 20 (1): 145-160.

[19] LIU H, GU F, ZHANG Q. Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems [J]. IEEE Transactions on Evolutionary Computation, 2014, 18(3): 450-455.

[20] 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.

[21] LI K, DEB K, ZHANG Q, et al. An evolutionary many-objective optimization algorithm based on dominance and decomposition [J]. IEEE Transactions on Evolutionary Computation, 2015, 19(5): 694-716.

[22] CAI X, MEI Z, FAN Z. A decomposition-based many-objective evolutionary algorithm with two types of adjustments for direction vectors [J]. IEEE Transactions on Cybernetics, 2018, 48(8): 2335-2348.

[23] TRIVEDI A, SRINIVASAN D, SANYAL K, et al. A survey of multiobjective evolutionary algorithms based on decomposition [J]. IEEE Transactions on Evolutionary Computation, 2017, 21(3): 440-462.

[24] CHENG R, JIN Y, OLHOFER M, et al. A reference vector guided evolutionary algorithm for many-objective optimization [J]. IEEE Transactions on Evolutionary Computation, 2016, 20(5): 773-791.

[25] ZHOU A, QU B, 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.

[26] GIAGKIOZIS I, FLEMING P J. Methods for multi-objective optimization: an analysis [J]. Information Sciences, 2015, 293: 338-350.

[27] DAS I, DENNIS J E. Normal-boundary intersection: a new method for generating the Pareto surface in nonlinear multi-criteria optimization problems [J]. SIAM Journal on Optimization, 1998, 8(3): 631-657.

[28] ZITZLER E, DEB K, THIELE L. Comparison of multiobjective evolutionary algorithms: empirical results [J]. Evolutionary Computation, 2000, 8(2): 173-195.

[29] DEB K, THIELE L, LAUMANNS M, et al. Scalable test problems for evolutionary multi-objective optimization [M]// ABRAHAM A, JAIN L, GOLDBERG R. Evolutionary Multiobjective Optimization: Theoretical Advances and Applications. London: Springer, 2005: 105-145.

[30] 郑金华,邹娟. 多目标进化优化[M]. 北京:科学出版社,2017: 203-208. (ZHENG J H, ZOU J. Multi-objective evolutionary optimization[M]. Beijing: Science Press, 2017: 203-208.)

[31] 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

[32] BOSMAN P A N, THIERENS D. The balance between proximity and diversity in multiobjective evolutionary algorithms [J]. IEEE Transactions on Evolutionary Computation, 2003, 7(2): 174-188.

[33] LI M, ZHENG J. Spread assessment for evolutionary multi-objective optimization [C]// Proceedings of the 2009 International Conference on Evolutionary Multi-Criterion Optimization, LNCS 5467. Berlin: Springer, 2009: 216-230.

[34] WANG Z K, ZHANG Q F, LI H, et al. On the use of two reference points in decomposition based multiobjective evolutionary algorithms [J]. Swarm and Evolutionary Computation, 2017, 34: 89-102.

This work is partially supported by the National Natural Science Foundation of China (61763026).

LI Erchao, born in 1980, Ph. D., professor. His research interests include artificial intelligence, multi-objective optimization, robot control.

LI Kangwei, born in 1994, M. S. candidate. His research interests include multi-objective optimization.