文章编号: 1001-0920(2017)09-1599-08 DOI: 10.13195/j.kzyjc.2016.0873

昂贵区间多目标优化空间数据挖掘求解策略

陈志旺, 赵子铮†, 姚嘉楠, 韩 艳

(燕山大学 工业计算机控制工程河北省重点实验室，河北 秦皇岛 066004)

摘 要: 针对优化函数未知的昂贵区间多目标优化问题,提出一种基于主曲线建模的NSGA-II算法. 该算法首先根据决策空间流形分布的种群数据构建K主曲线;然后利用所构建的K主曲线模型,通过插值和延展的方法生成子代. 与遗传算法的随机生成子代策略相比,通过所提出方法生成有效子代效率会更高. 由于目标空间拥挤距离无法求出,为此利用K主曲线找出待测解的前、后近距离解,按照决策空间拥挤距离对同序值解进行筛选,从而实现NSGA-II算法的改进.

关键词: 多目标优化；空间数据挖掘；区间规划；NSGA-II；主曲线

中图分类号: TP273 **文献标志码:** A

Spatial data mining strategy for expensive interval multi-objective optimization

CHEN Zhi-wang, ZHAO Zi-zheng†, YAO Jia-nan, HAN Yan

(Key Lab of Industrial Computer Control Engineering of Hebei Province，Yanshan University，Qinhuangdao 066004，China)

Abstract: In this paper, an improved NSGA - II algorithm is proposed based on the principal curve modeling for solving the expensive interval multi-objective optimization with unknown objective function. Firstly, the proposed algorithm builds a K principal curve using the population data of the manifold distribution in decision space. Then, a new offspring is generated through interpolation and extension according to the built K principal curve, and the proposed strategy of offspring generation is more efficient than that of random offspring generation in the genetic algorithm. Finally, because of the absence of the crowding distance in objective space, the closest solutions before and after the candidate solution can be found based on the built K principal curve, so the solutions with same sequence are screened by crowding distance in decision space, thus the NSGA-II is improved.

Keywords: multi-objective optimization；spatial data mining；interval programming；NSGA-II；principal curve

0 引 言

在现实生活、工程中, 人们经常会遇到使多个目标在给定区域同时最佳的优化问题,这些目标常常相互冲突,此类问题称作多目标优化问题(MOP)[1]. 近些年,采用进化算法求解MOP取得的成果较多[2],但求解的对象多是优化函数已知的确定性问题. 实际优化问题与此不同: 1) 优化函数经常是未知的,且评估实验成本非常昂贵,例如空气动力学优化设计[3]、药物设计[4] 等; 2) 目标函数中含有可用区间表示的不确定变量,如电力系统优化调度[5] 等. 本文将具有上述特点的问题总结为“优化函数未知的昂贵区间多目标优化问题”.

对于昂贵区间多目标优化问题,由于优化函数未知,无法利用优化函数在目标空间求解,即无法利用目标函数值的支配关系推动算法进化,需要在决策空间利用空间数据挖掘算法(如模式识别等)求解. Gong等[6-7]将决策者的偏好融入到区间多目标优化问题中,在建立区间优化问题偏好多面体的理论基础上,提出了一种基于偏好多面体的区间多目标交互式遗传算法. 文献[8]将支配性预测用于目标函数未知的区间多目标优化问题,并利用决策空间数据的流形方法[9-10]解决了无目标向量的情况下相同序值解

收稿日期: 2016-06-07；**修回日期:** 2016-10-08.

基金项目: 国家自然科学基金项目(61573305, 61403332)；河北省自然科学基金青年基金项目(F2014203099, F2015203400)；燕山大学青年教师自主研究计划课题(13LGA006).

作者简介: 陈志旺(1978−), 男, 副教授, 从事多目标优化、多属性决策等研究；赵子铮(1990−), 男, 硕士生, 从事多目标优化的研究.

†通讯作者. E-mail: 58052134@qq.com

的筛选.

上述在决策空间中求解多目标优化问题的策略,可概括为解决点(遗传个体)与点、点(遗传个体)与线流形的关系问题,因此都可以看作是对决策空间进行“空间数据挖掘”[11]. 空间数据挖掘利用几何技术方法从空间数据中挖掘潜在的规律,而多目标遗传算法的种群个体在决策和目标空间中拥有空间和距离的特性,且相互邻近的种群个体之间存在相互影响,进而种群个体之间关系更为复杂. 因此,可将多目标遗传算法看成一种特殊的空间数据挖掘方法[12],即对解空间中所有(点)个体的分布评估. 由文献[9-10]可知,一个连续m个目标的多目标优化问题的Pareto解集会呈现出一个分段连续的$m-1$维流形. Zhou等[13]根据决策空间呈现流形的数据分布,采用主成分分析算法在决策空间建立概率模型,并交替采用遗传操作和概率建模的策略来产生子代. Zhou等[9]采用了新的收敛评判准则,由该准则决定选择遗传操作或是通过建立概率模型产生子代. 文献[10]提出了基于规则模型的多目标分布估计算法(RM-MEDA). 文献[14]采用多重分形作为收敛准则评判决策空间流形标准,并提出建立非线性模型.

上述多目标优化算法,利用决策空间数据分布的规律和特征规则引导算法进化,但存在以下3方面问题: 1) 文献都以决策空间数据呈流形为依据建立模型,建模方法多采用聚类算法结合主成分分析(PCA)法,而没有考虑数据间存在的强非线性关系,因此建模不准确,且结果易受聚类数影响,文献[14]中虽提出建立非线性模型,但并未给出具体的方法和策略; 2) 文献中的模型进化算法全部运用在目标函数已知的多目标优化中,而对于优化函数未知的区间多目标问题少有运用; 3) 文献中的模型进化算法都需要用收敛准则指导算法是否运用模型生成子代,即未充分利用决策空间数据的规律引导算法进化.

针对以上问题. 本文基于文献[8],将流形建模用到优化函数未知的昂贵区间多目标问题中,即在决策空间中建立非线性主曲线模型,并在主曲线模型上运用子代生成算法产生子代. 对于决策空间拥挤距离的问题,通过引入决策空间的主曲线模型,解决了无目标向量情况下对相同序值解筛选的问题.

1 区间多目标优化问题基本概念

区间多目标优化问题如下:

$$\begin{aligned}&\min_{\boldsymbol{x}} \boldsymbol{F}(\boldsymbol{x},\boldsymbol{u})=(f_1(\boldsymbol{x},\boldsymbol{u}),f_2(\boldsymbol{x},\boldsymbol{u}),\cdots,f_m(\boldsymbol{x},\boldsymbol{u})),\\&\quad m=1,2,\cdots,z.\\&\text{s.t. } g_j(\boldsymbol{x},\boldsymbol{u})\geqslant a_j=[\underline{a_j},\overline{a_j}],\ j=1,2,\cdots,n;\\&h_k(\boldsymbol{x},\boldsymbol{u})=b_k=[\underline{b_k},\overline{b_k}],\ k=1,2,\cdots,n^*;\\&\boldsymbol{x}=(x_1,\cdots,x_q)\in\boldsymbol{R}^q,\ x_t\in[\underline{x_t},\overline{x_t}],\ t=1,2,\cdots,q;\\&\boldsymbol{u}=(u_1,\cdots,u_p)\in\boldsymbol{R}^p,u_l\in[\underline{u_l},\overline{u_l}],\ l=1,2,\cdots,p.\end{aligned}\tag{1}$$

其中: $g_j(\boldsymbol{x},\boldsymbol{u})$和$h_k(\boldsymbol{x},\boldsymbol{u})$均为区间函数,$g_j(\boldsymbol{x},\boldsymbol{u})\geqslant a_j$为第$j$个区间不等式约束,$h_k(\boldsymbol{x},\boldsymbol{u})=b_k$为第$k$个区间等式约束,其他参数见文献[15].

定义1[15] 令$P(f_m(\boldsymbol{x}_i,\boldsymbol{u})\leqslant f_m(\boldsymbol{x}_j,\boldsymbol{u}))$为区间$f_m(\boldsymbol{x}_i,\boldsymbol{u})$小于等于区间$f_m(\boldsymbol{x}_j,\boldsymbol{u})$的区间可能度,有

$$P(f_m(\boldsymbol{x}_i,\boldsymbol{u})\leqslant f_m(\boldsymbol{x}_j,\boldsymbol{u}))=\begin{cases}0,\ \overline{f_m(\boldsymbol{x}_j,\boldsymbol{u})}\leqslant\underline{f_m(\boldsymbol{x}_i,\boldsymbol{u})};\\ \dfrac{(\overline{f_m(\boldsymbol{x}_j,\boldsymbol{u})}-\underline{f_m(\boldsymbol{x}_i,\boldsymbol{u})})^2}{8\cdot f_m^r(\boldsymbol{x}_i,\boldsymbol{u})\cdot f_m^r(\boldsymbol{x}_j,\boldsymbol{u})},\\ \underline{f_m(\boldsymbol{x}_j,\boldsymbol{u})}<\underline{f_m(\boldsymbol{x}_i,\boldsymbol{u})}<\overline{f_m(\boldsymbol{x}_j,\boldsymbol{u})}<\overline{f_m(\boldsymbol{x}_i,\boldsymbol{u})};\\ \dfrac{f_m^c(\boldsymbol{x}_j,\boldsymbol{u})-\underline{f_m(\boldsymbol{x}_i,\boldsymbol{u})}}{2\cdot f_m^r(\boldsymbol{x}_i,\boldsymbol{u})},\\ \underline{f_m(\boldsymbol{x}_i,\boldsymbol{u})}<\underline{f_m(\boldsymbol{x}_j,\boldsymbol{u})}<\overline{f_m(\boldsymbol{x}_j,\boldsymbol{u})}\leqslant\overline{f_m(\boldsymbol{x}_i,\boldsymbol{u})};\\ 1-\dfrac{(\overline{f_m(\boldsymbol{x}_i,\boldsymbol{u})}-\underline{f_m(\boldsymbol{x}_j,\boldsymbol{u})})^2}{8\cdot f_m^r(\boldsymbol{x}_i,\boldsymbol{u})\cdot f_m^r(\boldsymbol{x}_j,\boldsymbol{u})},\\ \underline{f_m(\boldsymbol{x}_i,\boldsymbol{u})}\leqslant\underline{f_m(\boldsymbol{x}_j,\boldsymbol{u})}<\overline{f_m(\boldsymbol{x}_i,\boldsymbol{u})}<\overline{f_m(\boldsymbol{x}_j,\boldsymbol{u})};\\ \dfrac{\overline{f_m(\boldsymbol{x}_j,\boldsymbol{u})}-f_m^c(\boldsymbol{x}_i,\boldsymbol{u})}{2\cdot f_m^r(\boldsymbol{x}_j,\boldsymbol{u})},\\ \underline{f_m(\boldsymbol{x}_j,\boldsymbol{u})}\leqslant\underline{f_m(\boldsymbol{x}_i,\boldsymbol{u})}<\overline{f_m(\boldsymbol{x}_i,\boldsymbol{u})}\leqslant\overline{f_m(\boldsymbol{x}_j,\boldsymbol{u})};\\ 1,\ \overline{f_m(\boldsymbol{x}_i,\boldsymbol{u})}\leqslant\underline{f_m(\boldsymbol{x}_j,\boldsymbol{u})}.\end{cases}\tag{2}$$

定义2[15] 对于式(1)的两个解$\boldsymbol{x}_i$和$\boldsymbol{x}_j$,如果区间可能度$P(f_m(\boldsymbol{x}_i,\boldsymbol{u})\leqslant f_m(\boldsymbol{x}_j,\boldsymbol{u}))$均不小于0.5,且存在区间可能度$P(f_m(\boldsymbol{x}_i,\boldsymbol{u})\leqslant f_m(\boldsymbol{x}_j,\boldsymbol{u}))$大于0.5,则称$\boldsymbol{x}_i$支配$\boldsymbol{x}_j$,记为$\boldsymbol{x}_i\succ_P\boldsymbol{x}_j$,也可称$\boldsymbol{x}_j$被$\boldsymbol{x}_i$支配,记为$\boldsymbol{x}_j\prec_P\boldsymbol{x}_i$,即

$$\begin{aligned}\boldsymbol{x}_i\succ_P\boldsymbol{x}_j\Leftrightarrow\ &\forall m\in(1,2,\cdots,z):\\&P(f_m(\boldsymbol{x}_i,\boldsymbol{u})\leqslant f_m(\boldsymbol{x}_j,\boldsymbol{u}))\geqslant 0.5;\\&\exists m\in(1,2,\cdots,z):\\&P(f_m(\boldsymbol{x}_i,\boldsymbol{u})\leqslant f_m(\boldsymbol{x}_j,\boldsymbol{u}))>0.5.\end{aligned}\tag{3}$$

区间距离公式、可行解和Pareto支配性的最近邻预测见文献[8].

2 决策空间数据K主曲线建模

空间数据挖掘对于揭示空间数据规律起着重要作用. 在空间数据挖掘的方法中主成分分析方法应用广泛[16],但在数据集具有非线性特征时,主曲线比主成分分析法能更好地反映数据的流形[17]. 在当前多种类型主曲线中,K主曲线因其分段线性化[18-19]的特点和矢量量化[20]的思想得到广泛应用,因此

本文采用K主曲线对决策空间数据进行建模. 由Karush-Kuhn-Tucker条件[9-10]可知,决策空间的Pareto解集呈现为流形,可以利用主曲线对空间流形形式的数据点进行非线性建模,实现由空间点建立空间线的模型. 建模过程是利用多段线段逼近光滑的主曲线,达到建立模型的目的. 在建模数据获取方法上,第1代建模数据由算法初始化(随机生成子代)给出,以后每代建模数据采用父代同序值数据进行建模.

决策空间上的K主曲线建模示意图见图1. 图1中的点选自NSGA-II算法第1代子代中序值为1的9个个体,这些个体组成建模点集X',其具体数值为

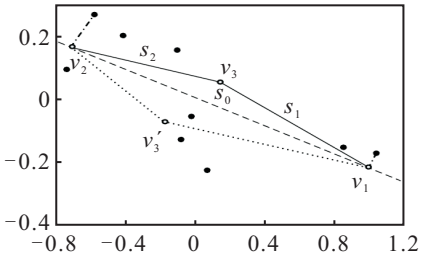
$$X' = [\boldsymbol{x}'_1, \boldsymbol{x}'_2, \cdots, \boldsymbol{x}'_i, \cdots, \boldsymbol{x}'_r] =$$

[0.678 9 2.818 8; 1.083 9 2.556 0; 2.155 2.440 4;

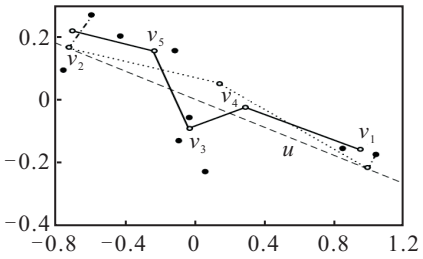
1.177 4 2.384 2; 1.961 0 2.457 6; 0.361 1 2.708 8;

0.998 7 2.771 4; 1.025 34 2.482 4; 0.518 9 2.885 7],

其中$r = 9$. 此处设每个子代解中有两个自变量,即$\boldsymbol{x}'_i = [x'_{i1}, x'_{i2}]$为二维向量.



(a) 生成初始主元曲线



(b) 调整顶点和增加顶点

图1　主曲线建模

图1中: "•"为决策空间同序值解$\boldsymbol{x}'_i$; "○"为主曲线顶点,用$\boldsymbol{v}_j$表示; "---"为第一主元方向; "–·–·"为投影方向; "○–○"为主曲线线段,用s_w表示; "○ ··· ○"为淘汰的主曲线. K主曲线建模主要包括如下部分:

1) 主元曲线生成. 首先采用"逐样本均值消减方法",对X'中的数据进行标准化处理. 该标准化方法能移除数据的统计平均值,体现数据的差异性,有

$$\boldsymbol{x}_{ij} = (\boldsymbol{x}'_{ij} - \overline{\boldsymbol{x}_j}),\ i = 1, 2, \cdots, r,\ j = 1, 2, \quad (4)$$

其中$\overline{\boldsymbol{x}_j} = \dfrac{1}{r}\sum_{i=1}^{r} x'_{ij}$. 处理后的数据集为$X = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_i, \cdots, \boldsymbol{x}_r]$.

利用PCA方法找到最大主元方$\boldsymbol{e}$,得到如图1(a)所示过原点且斜率为$\boldsymbol{e} = [e_1 \quad e_2]$的直线$s_0$,称为主元曲线. 计算数据点$\boldsymbol{x}_i = [x_{i1} \quad x_{i2}]$中个体到$s_0$上的投影值$x''_i$($x''_i$为标量),有

$$x''_i = \boldsymbol{e}\boldsymbol{x}_i^{\mathrm{T}} = [e_1 \quad e_2]\begin{bmatrix} x_{i1} \\ x_{i2} \end{bmatrix} = e_1 x_{i1} + e_2 x_{i2}. \quad (5)$$

在得到的投影中找到最大最小投影值$\max(x''_i)$和$\min(x''_i)$,利用下式找到图1(a)中的顶点$\boldsymbol{v}_1$和$\boldsymbol{v}_2$:

$$\boldsymbol{v}_1 = \max(x''_i)\boldsymbol{e} = [\max(x''_i)e_1 \quad \max(x''_i)e_2],$$

$$\boldsymbol{v}_2 = \min(x''_i)\boldsymbol{e} = [\min(x''_i)e_1 \quad \min(x''_i)e_2]. \quad (6)$$

2) K主曲线"剪枝"操作. 按下式将数据集合X通过直线s_0分为两类,直线上侧为第1类D_{s_0},其他为第2类$D_{s'_0}$:

$$D_{s_0} = \{\boldsymbol{x} | \boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{e} > 0,\ i = 1, 2, \cdots, r\},$$

$$D_{s'_0} = \{\boldsymbol{x} | \boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{e} \leqslant 0,\ i = 1, 2, \cdots, r\}. \quad (7)$$

D_{s_0}和$D_{s'_0}$数据集合重心分别为

$$\boldsymbol{v}_3 = \begin{cases} \dfrac{1}{N_{D_{s_0}}} \displaystyle\sum_{\boldsymbol{x}_i \in D_{s_0}} \boldsymbol{x}_i,\ N_{D_{s_0}} > 0; \\ (\boldsymbol{v}_1 + \boldsymbol{v}_2)/2,\ \text{otherwise}. \end{cases} \quad (8)$$

$$\boldsymbol{v}'_3 = \begin{cases} \dfrac{1}{N_{D_{s'_0}}} \displaystyle\sum_{\boldsymbol{x}_i \in D_{s'_0}} \boldsymbol{x}_i,\ N_{D_{s'_0}} > 0; \\ (\boldsymbol{v}_1 + \boldsymbol{v}_2)/2,\ \text{otherwise}. \end{cases} \quad (9)$$

其中: $N_{D_{s_0}}$为属于D_{s_0}类集合中的元素个数,$\boldsymbol{v}_3$和$\boldsymbol{v}'_3$为新生成K主曲线顶点. 这样便构造出最初的两条折线f(由$\boldsymbol{v}_1$、$\boldsymbol{v}_3$、$\boldsymbol{v}_2$组成的折线)和f'(由$\boldsymbol{v}_1$、$\boldsymbol{v}'_3$、$\boldsymbol{v}_2$组成的折线),如图1(a)所示. 取其中一条折线作为初始折线. 数据集X到折线f中各线段的距离为

$$d_f(\boldsymbol{x}_i, f) = \frac{1}{n}\frac{1}{j}\sum_{i=1}^{n}\sum_{w=1}^{\lambda-1} d_s(\boldsymbol{x}_i, s_w). \quad (10)$$

其中: $d_s(\boldsymbol{x}_i, s_w)$为数据点$\boldsymbol{x}_i$到线段$s_w$的欧氏距离,$s_w$为相邻两顶点$\boldsymbol{v}_m$和$\boldsymbol{v}_n$生成的线段(如图1(a)中$s_1$由顶点$\boldsymbol{v}_1$和$\boldsymbol{v}_3$组成的线段,$s_2$由顶点$\boldsymbol{v}_3$和$\boldsymbol{v}_2$组成的线段). 通过比较$d_f(\boldsymbol{x}_i, f)$和$d_f(\boldsymbol{x}_i, f')$的大小进行$f$和$f'$的取舍,若$d_f(\boldsymbol{x}_i, f) < d_f(\boldsymbol{x}_i, f')$,则保留$f$舍弃$f'$. 式(10)和$f$、$f'$的取舍过程体现了主曲线强调寻找通过数据分布的"中心线"的特点,即使用近似折线代替线性主成分线分析数据,以求出对称变量之间的近似折线,这样的折线从数据的中部通过,可以实现对数据的线性平均. 由实际数据计算得到$d_f(\boldsymbol{x}_i, f) < d_f(\boldsymbol{x}_i, f')$,所以保留$f$舍弃$f'$,见图1(a).

3) K主曲线调整顶点. 调整顶点过程如图1(b)所示. 根据数据点$\boldsymbol{x}_i$到所有顶点$\boldsymbol{v}_j$的欧氏距离的极小值进行分类,按照下式进行:

$$\min_{1\leqslant j\leqslant\lambda} d(\boldsymbol{x}_i,\boldsymbol{v}_j), \tag{11}$$

其中$d(,)$表示两点之间的欧氏距离. 若顶点的个数为λ,则数据点分为λ类,属于顶点$\boldsymbol{v}_j$的数据点可表示为$x_i\in D_{v_j}$. 调整后顶点位置计算公式为

$$\boldsymbol{v}_j=\frac{1}{N_{D_{vj}}}\sum_{\boldsymbol{x}_i\in D_{v_j}}\boldsymbol{x}_i, \tag{12}$$

其中$N_{D_{v_j}}$为集合D_{v_j}中$\boldsymbol{x}_i$元素个数. 调整后的顶点如图1(b)所示.

4) K主曲线增加顶点. 增加顶点过程如图1(b)所示. 根据$\boldsymbol{x}_i$到线段的欧氏距离分类到距离最小的线段(相邻的两个顶点为线段),即

$$\min_{1\leqslant j\leqslant\lambda-1} d_s(\boldsymbol{x}_i,s_w),\ w=1,2,\cdots,\lambda-1. \tag{13}$$

式(13)中$d_s(\boldsymbol{x}_i,s_w)$与(10)相同. 若线段的个数为$\lambda-1$,则数据点分为$\lambda-1$类,属于线段$s_w$的数据点可表示为$\boldsymbol{x}_i\in D_{s_w}$. 新增顶点计算公式为

$$\boldsymbol{v}_j=\frac{1}{N_{D_{s_w}}+2}\Big(\sum_{\boldsymbol{x}_i\in D_{s_w}}\boldsymbol{x}_i+\boldsymbol{v}_m+\boldsymbol{v}_n\Big),\ j\leqslant 2^k. \tag{14}$$

其中:k为算法循环次数,s_w为相邻两顶点$\boldsymbol{v}_m$和$\boldsymbol{v}_n$生成的线段. 由图1(b)和式(14)可知,第k次循环线段的增长数为2^k,而顶点会随着线段的增长而增长,因此顶点的增长规律也为2^k.

通过以上第3)和第4)步过程迭代,提取了决策空间数据的流形特征,并依据该特征采用K主曲线逼近数据流形,进而完成了空间数据模型的建立.

综上所述,主曲线建模步骤如下.

Step 1: 对数据X'由式(4)进行初始化,通过PCA算法获取数据X第一主成分方向$\boldsymbol{e}$,令循环次数k=1.

Step 2: 由式(5)计算数据点$\boldsymbol{x}_i(i=1,2,\cdots,r)$在$\boldsymbol{e}$上的投影值,由式(6)找到主曲线两个端点$\boldsymbol{v}_1$和$\boldsymbol{v}_2$.

Step 3: 按照式(7)将空间数据点分为两类集合D_{s_0}和$D_{s_0'}$.

Step 4: 利用式(8)和(9)计算$\boldsymbol{v}_3$和$\boldsymbol{v}_3'$,构建两个初始折线$f=\{\boldsymbol{v}_1,\boldsymbol{v}_3,\boldsymbol{v}_2\}$和$f'=\{\boldsymbol{v}_1,\boldsymbol{v}_3',\boldsymbol{v}_2\}$.

Step 5: 通过式(10)得出$d_f(\boldsymbol{x},f)$和$d_f(\boldsymbol{x},f')$,如果$d_f(\boldsymbol{x},f)<d_f(\boldsymbol{x},f')$,则保留$f$,删除$f'$.

Step 6: 依据式(12)调整现有的顶点位置.

Step 7: 按照式(14)计算新增顶点.

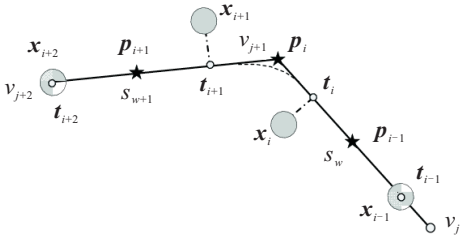
Step 8: 令$k=k+1$,如果$k<\log_2(r)$,则转至Step 6,否则算法停止.

上述算法中,Step 8每次增加新顶点个数为2^k,k为算法迭代次数. 因此采用$k<\log_2(r)$作为算法停止准则,使生成的顶点数小于数据点的个数,否则,若生成的顶点数大于数据点的个数,则会造成“有限信息的过拟合”.

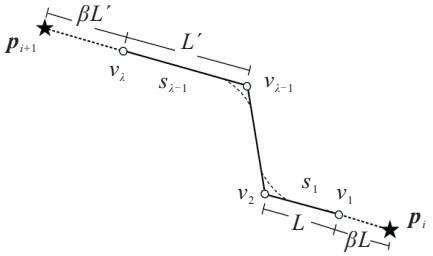
3 子代生成算法

空间数据建模是从存在的数据中找到一个函数模型,使该模型最好地描述这些已知的空间数据规律,并能根据函数模型求出区域范围内其他任意点的值[21]. 由第2节可知,多目标优化问题的决策空间解分布满足主曲线的规律,而文献[8]中NSGA-II多目标优化算法采用交叉变异方法生成具有随机性的子代,此方法未充分挖掘决策空间数据的规律来引导算法的进化. 本文利用K主曲线建立决策空间数据模型,利用K主曲线模型引导子代的生成,有利于提高算法的收敛速度.

子代生成算法如图2所示. 图2中:“$\star$”为产生的子代,用$\boldsymbol{p}_i$表示;L为线段长度;β为延展率.



(a) 内插策略



(b) 延展策略

图2 子代生成算法

子代生成算法首先获得同序值种群个体在K主曲线上的投影值. 例如,图2(a)中数据点$\boldsymbol{x}_i=[x_{i1},x_{i2}]$到$s_w$线段的投影点为$\boldsymbol{x}_i$到$s_w$线段的垂足$\boldsymbol{t}_i=[t_{i1},t_{i2}]$,$s_w$线段两端点为$[v_{j1},v_{j2}]$,$[v_{(j+1)1},v_{(j+1)2}]$. 根据平面几何垂足计算公式可得

$$\begin{aligned}t_{i1}=&\frac{\mu*(v_{(j+1)2}*v_{j1}-v_{j1}*v_{j2})}{v_{(j+1)1}-v_{j1}+\mu*v_{(j+1)2}-\mu*v_{j2}}+\\&\frac{v_{j2}*(v_{j2}-x_{i2})-v_{(j+1)2}*(v_{j2}-x_{i2})}{v_{(j+1)1}-v_{j1}+\mu*v_{(j+1)2}-\mu*v_{j2}}+\\&\frac{v_{(j+1)1}*x_{i1}-v_{j1}*x_{i1}}{v_{(j+1)1}-v_{j1}+\mu*v_{(j+1)2}-\mu*v_{j2}},\end{aligned} \tag{15}$$

$$t_{i2}=\mu*(t_{i1}-v_{j1})+v_{j2}, \tag{16}$$

其中$\mu=(v_{(j+1)2}-v_{j2})/(v_{(j+1)1}-v_{j1})$.

若数据点在主曲线上(如图2(a)中数据点$\boldsymbol{x}_{i-1}$和$\boldsymbol{x}_{i+2}$在K主曲线线段s_w上),则投影点和数据点重合,即$\boldsymbol{t}_{i-1}=\boldsymbol{x}_{i-1}$,$\boldsymbol{t}_{i+2}=\boldsymbol{x}_{i+2}$. 得到投影值后需对投影值进行排序,排序的方法是选择线段s_w的某个

端点$\boldsymbol{v}_j$为基准,将投影点与$\boldsymbol{v}_j$的距离称为投影距离,按照投影距离进行排序. 投影距离的公式为

$$d_i = d(\boldsymbol{t}_i, \boldsymbol{v}_j). \tag{17}$$

利用式(17)得到每段线段上所有的投影距离d_i,将d_i按升序排列. 由于d_i与$\boldsymbol{t}_i$存在对应关系,通过d_i可以得到$\boldsymbol{t}_i$的顺序. 重复上述过程得到每段线段上$\boldsymbol{t}_i$的顺序. 按照线段首尾相连的顺序确定各线段$\boldsymbol{t}_i$的先后顺序,这样即可得到主曲线上$\boldsymbol{t}_i$的顺序. 如图2(a)中的数据点$\boldsymbol{x}_{i-1}, \boldsymbol{x}_i, \boldsymbol{x}_{i+1}, \boldsymbol{x}_{i+2}$,由式(13)得到$\boldsymbol{x}_{i-1}, \boldsymbol{x}_i$属于线段$s_w$;$\boldsymbol{x}_{i+1}, \boldsymbol{x}_{i+2}$属于线段$s_{w+1}$. 由式(15)和(16)计算投影点,得到线段$s_w$上投影$\boldsymbol{t}_{i-1}, \boldsymbol{t}_i$;线段$s_{w+1}$上投影$\boldsymbol{t}_{i+1}, \boldsymbol{t}_{i+2}$. 由式(17)计算投影距离,在线段$s_w$上得到$\boldsymbol{t}_{i-1}$小于$\boldsymbol{t}_i$,在线段$s_{w+1}$上得到$\boldsymbol{t}_{i+1}$小于$\boldsymbol{t}_{i+2}$. 按照线段顺序$s_w$在$s_{w+1}$前,得到投影顺序$\boldsymbol{t}_{i-1}, \boldsymbol{t}_i, \boldsymbol{t}_{i+1}, \boldsymbol{t}_{i+2}$.

基于投影顺序,子代生成方法分为插值策略和延展策略:

1) 插值策略,见图2(a). 图中"$\star$"为产生的子代,在两相邻插值点$\boldsymbol{t}_i$和$\boldsymbol{t}_{i-1}$上取平均值产生子代,有

$$\boldsymbol{p}_{i-1} = [p_{(i-1)1} \quad p_{(i-1)2}] = \left[\frac{(t_{i1} + t_{(i-1)1})}{2} \quad \frac{(t_{i2} + t_{(i-1)2})}{2}\right]. \tag{18}$$

利用式(18)产生子代时,r个投影值可产生$r-1$个子代. 此外,线段连接的转折处作为子代保留,如图2(a)的$\boldsymbol{p}_i$所示.

2) 延展策略,见图2(b). 延展策略即在第1段线段s_1和最后一段$s_{\lambda-1}$的延长线段上,新生成延展线段的端点即为新产生的子代. 求得第一段和最后一段线段的距离,即

$$L = d(\boldsymbol{v}_2, \boldsymbol{v}_1),\ L' = d(\boldsymbol{v}_\lambda, \boldsymbol{v}_{\lambda-1}). \tag{19}$$

利用式(19)得到L和L',进而得到延展线段距离公式

$$H = \beta L,\ H' = \beta L'. \tag{20}$$

式(20)中β为延展率,延展率数值并不为一,采用不同的延展率会对Pareto解集的测度产生影响. 如图2(b)中,第1个和最后1个顶点为$\boldsymbol{v}_1 = [v_{11} \ v_{12}], \boldsymbol{v}_\lambda = [v_{\lambda 1} \ v_{\lambda 2}]$,设延展策略产生的子代为$\boldsymbol{p}_i = [p_{i1} \ p_{i2}]$, $\boldsymbol{p}_{i+1} = [p_{(i+1)1} \ p_{(i+1)2}]$,利用子代到顶点的欧氏距离等于$H$或$H'$,得到公式

$$d(\boldsymbol{p}_i, \boldsymbol{v}_1) = H,\ d(\boldsymbol{p}_{i+1}, \boldsymbol{v}_\lambda) = H'. \tag{21}$$

由式(20)和(21)推导出求解首段线段延展子代公式

$$p_{i1} = \sqrt{\frac{(\beta^2 * L^2)}{\varepsilon^2 + 1}} + v_{11},$$

$$p_{i2} = \varepsilon\left(\sqrt{\frac{(\beta^2 * L^2)}{\varepsilon^2 + 1}}\right) + v_{12}, \tag{22}$$

其中$\varepsilon = (v_{22} - v_{12})/(v_{21} - v_{11})$. 同理,末段线段延展子代公式可按式(22)推得. 利用以上公式完成$\boldsymbol{p}_i$和$\boldsymbol{p}_{i+1}$的求解,即完成延展策略算法.

综上,基于主曲线的插值策略和延展策略提高了生成优质子代的效率. 此外,延展策略增加了空间数据的搜索能力.

4　决策空间拥挤距离

对于优化函数未知的区间多目标优化问题,由于无法在目标空间计算拥挤距离,在决策空间采用拥挤距离思想对同序值解进行排序[8]. 第2节通过对决策空间数据进行挖掘,得到了描述决策空间数据流形的主曲线. 主曲线具有"自相合"特征,即主曲线上的每个点是所有投影至该点的数据点的期望均值,如果两个点$\boldsymbol{x}_i$和$\boldsymbol{x}_{i+1}$是相邻的,则可以推出在曲线上的投影$\boldsymbol{t}_i$和$\boldsymbol{t}_{i+1}$也是相邻的,因此保证了数据分布的序结构在主曲线上保持不变,进而可以通过主曲线上的投影确定种群解的前后近距离解. 利用此方法计算决策空间的拥挤距离,拥挤距离越大则解集在决策空间的多样性越好[22],这样便实现了相同序值解的筛选.

决策空间拥挤距离示意图如图3所示. 图3中,"--"为拥挤距离$\mathrm{dis}(\boldsymbol{x}_i)$,具体计算方法如下: 根据第3节方法找到$\boldsymbol{t}_i$的升序关系为$\boldsymbol{t}_{i-2}, \boldsymbol{t}_{i-1}, \boldsymbol{t}_i, \boldsymbol{t}_{i+1}, \boldsymbol{t}_{i+2}$,由于$\boldsymbol{t}_i$与$\boldsymbol{x}_i$之间存在对应关系,数据的升序为$\boldsymbol{x}_{i-2}, \boldsymbol{x}_{i-1}, \boldsymbol{x}_i, \boldsymbol{x}_{i+1}, \boldsymbol{x}_{i+2}$. 找出$\boldsymbol{x}_i$的前后近距离解,计算$\boldsymbol{x}_i$的拥挤距离为

$$\mathrm{dis}(\boldsymbol{x}_i) = \begin{cases} d(\boldsymbol{x}_{i+1}, \boldsymbol{x}_{i-1}),\ i = 2, 3, \cdots, r-1; \\ \mathrm{Inf},\ i = 1,\ i = r. \end{cases} \tag{23}$$

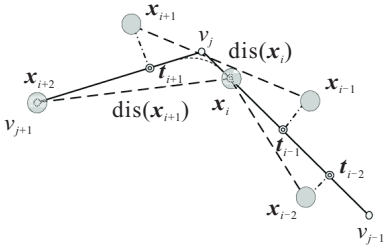


图3　决策空间拥挤距离

由于保留位于整体流形两端点处的解,可提高流形的散布性,图3中$\boldsymbol{x}_{i+2}$和$\boldsymbol{x}_{i-2}$可直接定为无穷大. 与文献[8]拥挤距离求解算法相比,本文算法省略了聚类步骤,并且利用多段折线而不是一段直线去逼近种群流形,因此本文算法精度更高.

5 改进的NSGA-II算法

文献[8]只利用遗传算法的随机性对优化问题求解,并未充分挖掘决策空间数据的规律,因此本文进行如下改进:

1) 采用K主曲线对决策空间种群非线性数据进行建模.

2) 根据K主曲线模型,采用内插策略和延展策略产生子代,代替传统遗传算法的交叉变异操作,即利用确定性操作代替随机操作.

3) 针对目标函数未知的情况下同序值解的排序问题,利用K主曲线模型确定前后近距离解,进而求得拥挤距离.

Step 1: 初始化算法参数. G为算法运行总代数,k^*为进化代数,并令$k^*=0$, ke为判断当前代为评估或预测环节的参数,$N(k^*)$为初始化种群,pop为种群规模. 同时,初始化算法必备集合: 可行解备选集、训练样本集和精英储备集.

Step 2: 利用第2节K主曲线建模策略,对父代决策空间种群进行主曲线建模,并运用子代生成算法(第3节)得到种群个数为pop的子代. 通过ke是否能整除k^*进行分支选择,若是则执行下一步,否则转至Step 6.

Step 3: 合并精英储备集、子代和父代得到新种群$R(k^*)$(种群规模不大于3× pop).

Step 4: 采用训练样本集策略[8]更新训练样本集.

Step 5: 通过评估结果,按照序值和目标空间拥挤距离对种群$R(k^*)$进行筛选,进而得到种群$N(k^*+1)$,同时更新精英储备集. 转至Step 11.

Step 6: 通过合并父代、子代构成新的种群$R(k^*)$.

Step 7: 通过ke是否能整除k^*-1进行分支选择,若是则执行下一步,否则转至Step 9.

Step 8: 应用可行解备选集策略[8]去除不满足约束条件的解.

Step 9: 采用最近邻法预测[8]获得种群$R(k^*)$中个体的序值,序值小者为优.

Step 10: 利用第4节方法对决策空间同序值解进行拥挤距离排序,进而得到新种群$N(k^*+1)$.

Step 11: 判断k^*是否等于G,如果等于则转至Step 12,否则令$k^*=k^*+1$,转至Step 2.

Step 12: 输出最优Pareto解集.

6 仿真优化实验

以优化问题Q_1[23]和Q_2[15]为例,验证所提出算法的有效性.

Q_1:

$$\begin{aligned}&\min f_1(\boldsymbol{x},\boldsymbol{u})=-10\mathrm{e}^{-u_1\sqrt{x_1^2+x_2^2}},\\&\quad f_2(\boldsymbol{x},\boldsymbol{u})=|x_1|^{0.8}+|x_2|^{0.8}+u_2(\sin x_1^3+\sin x_2^3),\\&\quad x_1,x_2\in[-5,5],u_1\in[0.19,0.21],\ u_2\in[4.9,5.1];\end{aligned}\tag{24}$$

Q_2:

$$\begin{aligned}&\min\ f_1(\boldsymbol{x},\boldsymbol{u})=u_1(x_1+x_2-7.5)^2+\\&\qquad u_2^2(x_2-x_1+3)^2/4,\\&\quad f_2(\boldsymbol{x},\boldsymbol{u})=u_1^2(x_1-1)/4+u_2^3(x_2-4)^2/2.\\&\text{s.t. }g_1(\boldsymbol{x},\boldsymbol{u})={u_1}^2(x_1-2)^3/2+u_2x_2-2.5\leqslant[0,0.3],\\&\quad g_2(\boldsymbol{x},\boldsymbol{u})=u_1^3x_2+u_2^2x_1-3.85-\\&\qquad 8u_2^2(x_2-x_1+0.65)^2\leqslant[0,0.3],\\&\quad x_1\in[0,5],\ x_2\in[0,3],\ u_1,u_2\in[0.9,1.1].\end{aligned}$$

针对目标函数为区间数这一特点,本文选用3种测度以检验所提出改进NSGA-II的性能[15]. 其中: E测度值越小,Pareto前沿的分布越均匀; D测度越大,分布越宽广; C测度越小,集合A趋近于集合B的程度越高. 以下实验数据无特殊说明,均为算法独立运行30次误差在一定范围内获得的结论. 算法参数取值如下: η初值为40, $\theta_\delta^{\max}=0.5$, pop$=40$, ke$=20$, $G=200$.

6.1 延展率对算法的影响

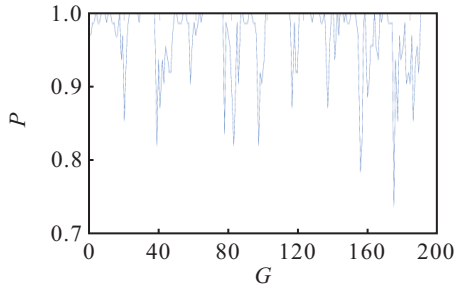
延展率β对Pareto前沿性能影响如表1所示. 由表1可知,延展率对性能指标的影响与具体的优化问题有关,并非越大或越小越好. 本文为了便于比较,仿真均取$\beta=0.2$.

表1 不同延展率下的前沿性能指标

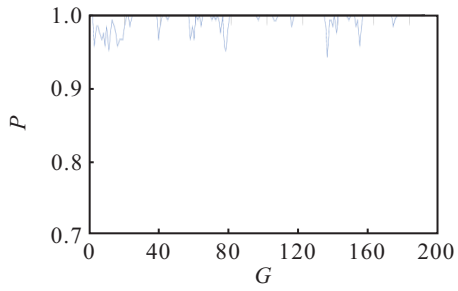
	延展率	$\beta=0.2$	$\beta=0.4$	$\beta=0.6$	$\beta=0.8$	$\beta=1$
Q_1	E测度	0.1313	0.1605	0.1849	0.1764	0.1795
	D测度	8.2871	8.1513	8.2608	8.2164	8.3559
	C测度	0.3386	0.3717	0.3735	0.3904	0.3750
Q_2	E测度	0.2952	0.3021	0.3142	0.3069	0.3148
	D测度	12.3704	12.4012	12.3274	12.4520	12.5021
	C测度	0.3004	0.3145	0.3247	0.3252	0.3106

6.2 决策空间拥挤距离筛选准确率

图4分别为Q_1、Q_2筛选准确率随遗传代数增加的走势图,表2为筛选准确率[8]平均值. 由上述数据可知,利用主曲线建模算法得出拥挤距离筛选准确率高于原始算法的筛选准确率,并且多次测试的平均值均大于0.9,验证了主曲线建模算法解决决策空间拥挤距离的有效性.



(a) Q_1的决策空间拥挤距离筛选准确率



(b) Q_2的决策空间拥挤距离筛选准确率

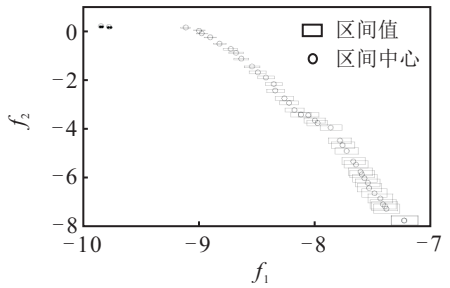
图4　拥挤距离预测准确率

表2　筛选准确率平均值

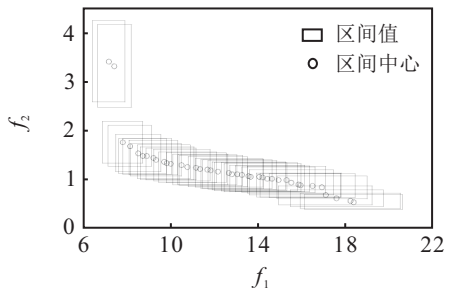
	Q_1	Q_2
聚类和PCA算法	0.866 0	0.982 0
K主曲线建模算法	0.962 7	0.996 1

6.3　本文算法的Pareto前沿与其他算法的比较

图5为采用主曲线建模和子代生成算法改进的NSGA-II算法,对问题Q_1、Q_2进行优化所得前沿. 由图5可见,得到的前沿具有较好的均匀性和散布度.



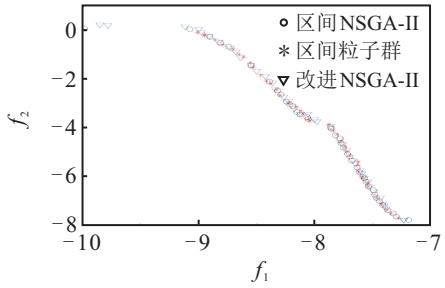
(a) Q_1改进NSGA-II的Pareto前沿



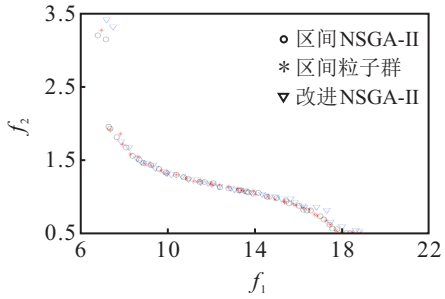
(b) Q_2改进NSGA-II的Pareto前沿

图5　改进算法的Pareto前沿

图6为文献[8]区间NSGA-II算法、文献[23]区间粒子群和本文改进算法的前沿中心对比图,表3为各算法的前沿测度指标. 由上述数据可知,3种算法的Pareto前沿分布非常接近,本文算法Q_1的D测度和E测度好于前二者,Q_2的E测度劣于前二者,但D测度好于前二者,总体测度优于其他两种算法.



(a) Q_1不同算法的Pareto前沿中心



(b) Q_2不同算法的Pareto前沿中心

图6　不同算法的Pareto前沿

表3　不同算法的测度指标

	NSGA-II[8]		区间粒子群[23]		本文NSGA-II	
	E测度	D测度	E测度	D测度	E测度	D测度
Q_1	0.160 9	8.249 8	0.145 7	8.176 8	0.131 3	8.287 1
Q_2	0.223 2	11.813 0	0.223 6	11.929 2	0.295 2	12.370 4

本文与文献[8]算法进行比较,两者同在优化函数未知的情况下进行优化求解,最终部分性能指标优于后者,体现了本文建模算法的可行性. 本文与文献[23]算法进行比较,后者采用区间粒子群算法在优化函数已知的情况下求解,本文算法采用区间NSGA-II算法在优化函数未知的情况下求解(优化函数未知情况更符合实际). 由于本文的已知条件劣于文献[23]算法,但最终性能指标与文献[23]接近,并且部分测度要优于优化函数已知算法,从而体现了本文比文献[23]的算法更具有实用性.

7　结　论

本文基于空间数据挖掘的思想,以决策空间数据呈现流形为前提,利用K主曲线建模得到种群个体决策空间分布规律. 在主曲线上运用插值和延展策略产生子代,代替文献[8]采用的交叉变异方法,提高算法效率,加速算法收敛. 同时在决策空间,利用主曲线模型得到前后近距离解,从而解决了决策空间拥挤距离计算的问题,提高了种群多样性,而且提高了拥挤距离在决策空间的筛选准确率.

参考文献(References)

[1] Sadollah A, Eskandar H, Kim J H. Water cycle algorithm for solving constrained multi-objective optimization problems[J]. Applied Soft Computing, 2015, 27(2): 279-298.

[2] 左兴权, 王春露, 赵新超. 一种结合多目标免疫算法和线性规划的双行设备布局方法[J]. 自动化学报, 2015, 41(3): 528-540.
(Zuo X Q, Wang C L, Zhao X C. Combining multi-objective immune algorithm and linear programming for double row layout problem[J]. Acta Automatica Sinica, 2015, 41(3): 528-540.)

[3] Jin Y, Sendhoff B. A systems approach to evolutionary multi-objective structural optimization and beyond[J]. IEEE Computational Intelligence Magazine, 2009, 4(3): 62-76.

[4] Douguet D. e-LEA3D: A computational-aided drug design web server[J]. Nucleic Acids Research, 2010, 38(2): 615-621.

[5] Li Y Z, Wu Q H, Jiang L, et al. Optimal power system dispatch with wind power integrated using nonlinear interval optimization and evidential reasoning approach[J]. IEEE Trans on Power Systems, 2016, 31(3): 2246-2254.

[6] Gong D W, Sun J, Ji X F, Evolutionary algorithms with preference polyhedron for interval multi-objective optimization problems[J]. Information Sciences, 2013, 233(1): 141-161.

[7] Gong D W, Ji X F, Sun J, et al. Interactive evolutionary algorithms with decision-maker's preferences for solving interval multi-objective optimization problems[J]. Neurocomputing, 2014, 137(4): 241-251.

[8] 陈志旺, 白锌, 杨七, 等. 区间多目标优化中决策空间约束、支配及同序解筛选策略[J]. 自动化学报, 2015, 41(12): 2115-2124.
(Chen Z W, Bai X, Yang Q, et al. Strategy of Constraint, Dominance and Screening Solutions with Same Sequence in Decision Space for Interval Multi-objective Optimizat- ion[J]. Acta Automatica Sinica, 2015, 41(12): 2115-2124.)

[9] Zhou A M, Jin Y C, Zhang Q F, et al. Combining model-based and genetics-based offspring generation for multi-objective optimization using a convergence criterion[C]. Proc of IEEE Congress on Evolutionary Computation(CEC). Vancouver: IEEE, 2006: 892-899.

[10] Zhang Q F, Zhou A M, Jin Y C. RM-MEDA: A regularity model-based multi-objective estimation of distributionalgorithm[J]. IEEE Trans on Evolutionary Computation, 2008, 12(1): 41-63.

[11] Li D R, Wang S L, Yuan H N, et al. Software and applications of spatial data mining[J]. Data Mining and Knowledge Discovery, 2016, 6(3): 84-114.

[12] Jiang Q, Lin H, Li J, et al. The research on spatial data mining module based on multi-objective optimization model for decision support system[J]. IEEE Computer Society on Second Wri Global Congress on Intelligent Systems. 2010, 4(2): 299-302.

[13] Zhou A M, Zhang Q F, Jin Y C, et al. A model-based evolutionary algorithm for bi-objective optimization [C]. The 2005 IEEE Congress on Evolutionary Computation. 2005, 3: 2568-2575.

[14] Zhang D M, Gong X S, Dai G M. Multi-objective evolutionary algorithm for principal curve model based on multifractal[J]. J of Computer Research & Development, 2011, 48(9): 1729-1739.

[15] 陈志旺,陈林. 求解约束多目标区间优化问题的改进NSGA-II[J]. 小型微型计算机系统, 2014, 35(11): 2502-2506.
(Chen Z W, Chen L. Improved NSGA-II for constrained multi-objective optimization problems with interval numbers[J]. J of Chinese Computer Systems, 2014, 35(11): 2502-2506.)

[16] Bhargava R, Singh P, Tanwar P S, et al. Spatial data mining using PCA[J]. Int J of Advanced Research in Computer Science, 2013, 4(7): 201-209.

[17] Zhang H, Pedrycz W, Miao D, et al. From principal curves to granular principal curves[J]. IEEE Trans on Cybernetics, 2014, 44(6): 748-760.

[18] Verbeek J J, Vlassis N, Kröse B. A k-segments algorithm for finding principal curves[J]. Pattern Recognition Letters, 2002, 23(2): 1009-1017.

[19] Aliyari Ghassabeh Y, Rudzicz F. Incremental algorithm for finding principal curves[J]. Iet Signal Processing, 2015, 9(7): 521-528.

[20] 李娟, 王宇平. 基于样本密度和分类误差率的增量学习矢量量化算法研究[J]. 自动化学报, 2015, 41(6): 1187-1200.
(LI J, Wang Y P. An incremental learning vector quantization algorithm based on pattern density and classication error ratio[J]. Acta Automatica Sinica, 2015, 41(6): 1187-1200.)

[21] 周晓根, 张贵军, 郝小虎. 局部抽象凸区域剖分差分进化算法[J]. 自动化学报, 2015, 41(7): 1315-1327.
(Zhou X G, Zhang G J, Hao X H. Diffe- rential Evolution Algorithm with Local Abstract Convex Region Partition[J]. Acta Automatica Sinica, 2015, 41(7): 1315-1327.)

[22] Xia T L, Zhang S H. An improved non-dominated sorting genetic algorithm for multi-objective optimization based on crowding distance[J]. Computational Intelligence, Networked Systems and Their Applications, 2014, 462(2): 66-76.

[23] 张勇, 巩敦卫, 郝国生, 等. 含区间参数多目标系统的微粒群优化算法[J]. 自动化学报, 2008, 34(8): 921-928.
(Zhang Y, Gong D W, Hao G S, et al. Partice swarm optimization for Multi-objective systems with interval parameters[J]. Acta Automatica Sinica, 2008, 34(8): 921-928.)

（责任编辑：郑晓蕾）