

控制与决策

Control and Decision

基于改进集成学习分类的代理辅助进化算法

顾清华, 张晓玥, 陈露

引用本文:

顾清华, 张晓玥, 陈露. 基于改进集成学习分类的代理辅助进化算法[J]. *控制与决策*, 2022, 37(10): 2456–2466.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2021.0388>

您可能感兴趣的其他文章

Articles you may be interested in

基于不确定度采样准则的费时问题优化算法

An optimization algorithm with uncertainty-based sampling strategy for expensive problems

控制与决策. 2022, 37(6): 1541–1549 <https://doi.org/10.13195/j.kzyjc.2020.1798>

基于向量角分解的高维多目标进化算法

Many-objective evolutionary algorithm based on vector angle decomposition

控制与决策. 2021, 36(3): 761–768 <https://doi.org/10.13195/j.kzyjc.2019.0925>

基于正态云模型的状态转移算法求解多目标柔性作业车间调度问题

State transition algorithm based on normal cloud model for solving multi-objective flexible job shop scheduling problem

控制与决策. 2021, 36(5): 1181–1190 <https://doi.org/10.13195/j.kzyjc.2019.1233>

一种基于MOEA/D的组合权重方法

A combination weight method based on MOEA/D

控制与决策. 2021, 36(12): 3056–3062 <https://doi.org/10.13195/j.kzyjc.2020.0592>

基于迁移学习灰支持向量回归机的交互式进化计算

Interactive evolutionary computation based on transfer learning grey support vector regression

控制与决策. 2021, 36(10): 2399–2408 <https://doi.org/10.13195/j.kzyjc.2020.0420>

基于改进集成学习分类的代理辅助进化算法

顾清华[†], 张晓玥, 陈露

(1. 西安建筑科技大学 管理学院, 西安 710055; 2. 西安市智慧工业感知计算与决策重点实验室, 西安 710055)

摘要: 当使用代理辅助进化算法求解昂贵高维多目标优化问题时,代理模型通常用于近似昂贵的适应度函数.然而,随着目标数的增加,近似误差将逐渐累积,计算量也会急剧增加.对此,提出一种基于改进集成学习分类的代理辅助进化算法,使用一种改进的装袋集成学习分类器作为代理模型.首先,从被昂贵的适应度评价的个体中选择一组分类边界,将所有个体分成两类;其次,利用这些带有分类标签的个体训练分类器,以对候选个体的类别进行预测;最后,选择有前途的个体进行昂贵适应度评价.实验结果表明,算法中所提出的代理模型可有效提高基于分类的代理辅助进化算法求解昂贵高维多目标优化问题的能力,且与目前流行的代理辅助进化算法相比,基于改进集成学习分类的代理辅助进化算法更具竞争力.

关键词: 昂贵高维多目标优化; 代理辅助进化算法; 代理模型; 集成学习; 装袋法; 分类器

中图分类号: TP273 文献标志码: A

DOI: 10.13195/j.kzyjc.2021.0388

引用格式: 顾清华, 张晓玥, 陈露. 基于改进集成学习分类的代理辅助进化算法[J]. 控制与决策, 2022, 37(10): 2456-2466.

Improved ensemble learning classification based surrogate-assisted evolutionary algorithm

GU Qing-hua[†], ZHANG Xiao-yue, CHEN Lu

(1. School of Management, Xi'an University of Architecture and Technology, Xi'an 710055, China; 2. Xi'an Key Laboratory of Smart Industry Perception Computing and Decision Making, Xi'an 710055, China)

Abstract: When using surrogate-assisted evolutionary algorithm to solve the expensive many-objective optimization problems, the surrogate is usually used to approximate the expensive fitness function. However, with the increase of the number of objectives, the approximation error will accumulate gradually and the amount of calculation will increase sharply. In order to solve this problem, we propose an improved ensemble learning classification based surrogate-assisted evolutionary algorithm, which uses an improved bagging ensemble as the surrogate. Firstly, a set of classification boundary individuals are selected from the individuals evaluated by the expensive fitness function, and the individuals are divided into two groups. Then, these individuals with the group labels are used to train a classifier to predict the groups of the candidate individuals. Finally, the promising individuals are selected to be evaluated by the expensive fitness function. The experimental results show that the proposed surrogate in the algorithm effectively improves the ability of the classification based surrogate-assisted evolutionary algorithm to solve the expensive many-objective optimization problems, and compared with the current popular surrogate-assisted evolutionary algorithms, the proposed improved ensemble learning classification based surrogate-assisted evolutionary algorithm is more competitive.

Keywords: expensive many-objective optimization; surrogate-assisted evolutionary algorithm; surrogate; ensemble learning; bagging; classifier

0 引言

高维多目标优化问题是具有大于 3 个目标的多目标优化问题,往往不存在单一的最优解,而是一组帕累托最优解^[1].进化算法因其具有较好的全局搜索

能力,且一次运行可以得到一组解,常被用于解决多目标优化问题^[2].在高维多目标优化问题中,由于帕累托前沿的选择压力削弱,需要进化算法进行成千上万次的适应度评价来对个体进行寻优.然而,在解

收稿日期: 2021-03-08; 录用日期: 2021-07-05.

基金项目: 国家自然科学基金项目(51774228, 51974223, 52074205); 陕西省自然科学基金杰出青年基金项目(2020JC-44).

责任编辑: 巩敦卫.

[†]通讯作者. E-mail: qinghuagu@126.com.

决许多实际的高维优化问题,如航空设计优化^[3]、药物设计优化^[4]时,适应度评价的计算成本非常高. 解决这类问题的常见方法是在进化算法中引入代理模型,这种方法被称为代理辅助进化算法(surrogate-assisted evolutionary algorithm, SAEA)^[5-6]. 根据代理模型的不同功能,大致可将SAEAs分为两类^[7].

第1类SAEAs是使用代理模型来近似候选个体的适应度值^[8-9]. 到目前为止,已有大量的研究致力于建立这类代理模型,包括多项式回归模型(polynomial regression)^[10]、克里金模型(Kriging)^[11]和神经网络模型^[12]等. 例如,文献[13]使用神经网络模型辅助进化,文献[14]使用克里金模型来近似每个子问题的目标函数. 但当这类SAEAs用来求解昂贵高维多目标优化问题时,近似误差将随着目标数量的增加而逐渐累积,计算量也将急剧增加. 第2类SAEAs采用分类器作为代理模型来过滤候选个体中较差的个体,因而SAEAs可以不需要建立用于近似每个目标函数的精确代理模型^[15]. 当新个体被这类代理模型预测时,得到的不再是一个具体的适应度值,而只是一个简单的标签,用于判断新个体的好坏,这类SAEAs通过样本点训练一个分类器来对新产生的个体进行分类. 2015年,文献[15]提出了一种基于分类和帕累托支配的多目标进化算法(CPS-MOEA),在CPS-MOEA中,个体被分为两类,分别被标记为正、负;然后,利用分类回归树(classification and regression tree, CART)对新生成的后代的类别进行预测,以减少评价子代的次数. 2018年,文献[16]提出了一种基于分类的代理辅助进化算法(CSEA),该算法使用前馈神经网络(feedforward neural network, FNN)预测候选个体与参考个体之间的支配关系,同时考虑到代理模型分类结果的不确定性,采用了一种可信度格局方法来判断所构建的代理模型是否准确. 这类基于分类的SAEAs用于求解问题时,往往只需建立一个代理模型来判断个体间的支配关系,相比于第1类SAEAs,大大减少了算法的计算代价.

除上述两种SAEAs中使用的分类回归树和人工神经网络外,现有分类器还包括支持向量机(support vector machine, SVM)^[17]、径向基函数网络(radial basis function network, RBF)^[18]等. 然而,用于建立代理模型的训练数据是稀缺且具有随机性的,导致所建立的这类代理模型的精度很低,进而使得算法容易向错误的方向收敛. 人们很难选择一个合适的分类器来辅助进化算法找到一组最优解. 因此,一个自然的想法是将多个代理模型组合在一起,以获得比

单一代理模型更高的分类精度^[19-20]. 文献[21]提出了一种双代理辅助的协同粒子群优化算法,使用一个上层全局代理模型和一个下层局部代理模型来寻求多个最优解,实验表明算法通过使用多个代理模型获得了很好的性能. 基于此,本文提出一种基于改进集成学习分类的代理辅助进化算法(improved ensemble learning classification based surrogate-assisted evolutionary algorithm, IECSEA)用于求解昂贵高维多目标优化问题,算法通过一种改进的装袋集成方法来构造代理模型,再根据分类结果的可信度,从子代个体中选择有前途的个体进行昂贵适应度评价.

本文的主要贡献如下:

- 1) 采用基于指标的逐个选择策略从昂贵适应度评价的个体中选择一组分类边界个体;
- 2) 提出一种改进的装袋集成分类器作为代理模型,用于将候选个体分成两组;
- 3) 采用双归档集管理策略进行模型管理.

1 相关技术

1.1 集成学习

集成学习可以组合一组弱分类器,使其成为一个强分类器. 集成学习方法已被证明在准确性和鲁棒性方面比单一模型更优^[22]. 提升法(boosting)^[23]和装袋法(bagging)^[24]是生成集成模型的两种常用方法. 前者采用序列化的思想,通过增加前一个弱分类器预测错误样本的权重来使后一个弱分类器更加关注预测错误样本,从而尽可能地修正这些错误. 后者使用并行的思想,即弱分类器的训练并行进行,给定一个包含 m 个样本集的初始样本集,使用有放回采样方法从初始样本集中随机选择一个样本;然后将提取的样本放回初始样本集中,以便在下一次采样中仍然可以选择该样本. 这样,在 m 次随机抽样之后,就可以获得包含 m 个样本的样本集,并得到分别包含 m 个训练样本的 T 个样本集,用 T 个样本集分别训练 T 个弱分类器,再将这些弱分类器进行组合.

1.2 逐个选择策略

逐个选择策略根据收敛性指标和分布性指标在种群中选择个体^[25]. 首先根据收敛性指标将个体进行排序,收敛性指标如下式所示:

$$c(x) = \sum_{m=1}^M f_m(x). \quad (1)$$

其中: $c(x)$ 为所有目标的总和, M 为总目标个数. 每次仅选择收敛性指标最好的一个个体,每当个体被选择,就利用小生境技术删除与其相似的个体^[26]. 其次,为了有效地评估高维目标空间中个体之间的距离,该

策略采用下式所示的基于余弦相似度的分布指标:

$$d(x_i) = (d_1(x_i), \dots, d_{|Q|}(x_i)), i = 1, \dots, |Q|, \quad (2)$$

$$d_j(x_i) = 1 - \cos \theta_{ij}, j = 1, \dots, |Q|, \quad (3)$$

$$\cos \theta_{ij} =$$

$$\frac{\sum_{m=1}^M (f_m(x_i) - z_m^*) \cdot (f_m(x_j) - z_m^*)}{\sqrt{\sum_{m=1}^M (f_m(x_i) - z_m^*)^2} \cdot \sqrt{\sum_{m=1}^M (f_m(x_j) - z_m^*)^2}}, \quad (4)$$

$$z_m^* = \min_{x_i \in Q} f_m(x_i), \quad (5)$$

其中 $d_j(x_i)$ 为 x_i 与 x_j 之间的距离. 余弦相似度采用两个向量之间的余弦值来估算它们的相似度. 当两个向量的方向完全相同时, 它们的余弦相似度为1; 当两个向量相互垂直时, 余弦相似度为0. 无论空间的维度为多少, 余弦相似度的范围都为[0,1]. 因此, 余弦相似度非常适用于表示高维空间中个体的相似度. 通过逐个选择策略, 可以在种群中得到一组具有良好的收敛性和分布性的个体, 且计算量很小.

1.3 可信度格局方法

文献[16]提供了一种可信度格局方法, 用于代理辅助个体的选择, 该方法使用测试集来评估基于分类的代理辅助进化算法中代理模型的不确定性, 并能在评估后给出一种个体的选择方案. 首先, 使用测试集分别计算 A 组个体的预测误差 m_1 和 B 组个体的预测误差 m_2 , 分类误差计算公式为

$$MAE = \frac{\sum_{i=1}^{|Q_c|} |(c - C_{p_i})|}{|Q_c|}. \quad (6)$$

其中: Q_c 是 c 类个体的集合, $|Q_c|$ 表示 Q_c 中所包含的个体个数, C_{p_i} 是模型预测的个体类别.

其次, 通过联合使用测试集上的误差 m_1 与 m_2 来评估模型预测的可靠性. 其具体流程为: 首先确定点 $P(m_1, m_2)$ 在可信度格局中所在的区域, 若点 P 位于区域 **R1**, 表示模型能正确预测 A 组的个体, 则选择 A 组中的个体生成子代, 直到达到终止条件; 若点 P 位于区域 **R2**, 表示模型既不能正确预测 A 组个体, 也不能正确预测 B 组个体, 导致分类模型不能提供关于候选个体真实类别的任何有用信息, 则此时不选择任何个体; 若点 P 位于区域 **R3**, 则表示 B 组个体可能被错误地预测为 A 组个体, 此时使用 B 组中的个体生成子代, 直到达到终止条件. 可信度格局的区域位置划分如图1所示, 其中 tr 的计算公式为

$$tr = \min\{rr, 1 - rr\}, \quad (7)$$

rr 表示 A 组个体的比例.

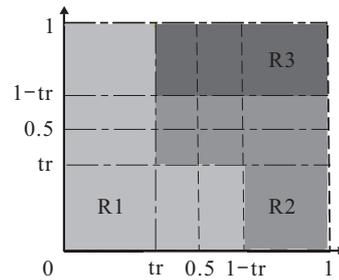


图1 可信度格局(图来源于文献[15])

2 基于改进集成学习分类的代理辅助进化算法

本文提出一种基于改进集成学习分类的代理辅助进化算法, 称作 **IECSEA**. 其中改进的集成学习分类器用于辅助进化算法选择用于昂贵适应度评价的个体. 图2为算法流程框架, 算法主要由分类边界选择、模型训练、代理辅助选择、归档集维护4部分组成. 算法1介绍了 **IECSEA** 的主要流程.

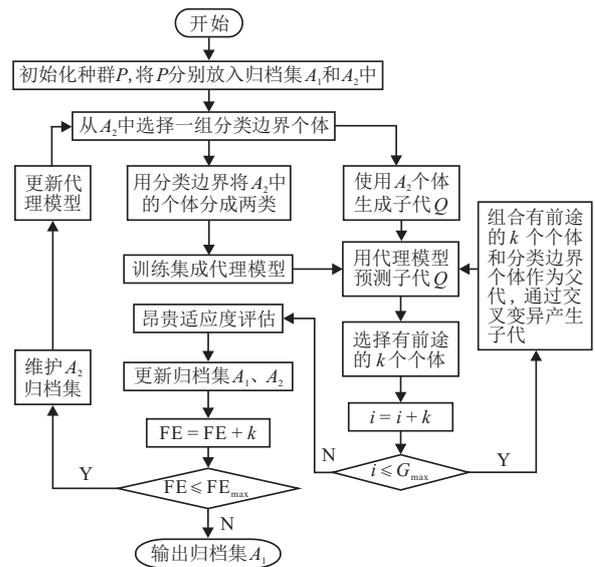


图2 IECSEA 流程框架

算法1 IECSEA 算法.

输入: 种群大小 N , 分类边界个体数 k , 昂贵适应度评价次数 FE_{max} , 代理模型更新前的最大预测数 G_{max} .

step 1: 采用拉丁超立方采样生成种群大小为 N 的初始种群 P , 并进行昂贵适应度评价;

step 2: 将初始种群 P 分别放入归档集 A_1 和 A_2 中;

step 3: 设定评价次数 $FE = N$;

step 4: while $FE \leq FE_{max}$;

step 5: 通过逐个选择策略从 A_2 中选择 k 个分类

边界个体;

step 6: 通过分类边界和分类准则将 A_2 中的个体分成两组;

step 7: 将分好类的个体划分为训练集和测试集;

step 8: 使用训练集训练改进的集成学习模型;

step 9: 使用测试集数据分别计算两组个体分类的错误率 m_1, m_2 ;

step 10: 判断点 $P(m_1, m_2)$ 位于可信度格局中的区位;

step 11: 使用模拟二进制交叉和多项式变异生成子代个体 Q ;

step 12: 设 $i = 0$;

step 13: while $i \leq G_{\max}$;

step 14: 使用代理模型判断子代个体类别;

step 15: 更新 $i = i + |k|$;

step 16: 选择有前途的 k 个体;

step 17: end while

step 18: 使用昂贵的适应度函数评价;

step 19: 更新 A_1 和 A_2 归档集;

step 20: 更新 $FE = FE + k$;

step 21: end while

step 22: 输出: A_1 归档集.

算法1中, 使用拉丁超立方采样生成分布良好的初始种群 P , 并对初始种群 P 进行昂贵适应度评价, 将它们放入归档集 A_1 和 A_2 中. 在主循环中, 首先从 A_2 归档集中选择 k 个个体作为分类边界, 将 A_2 中的个体分为两组, 再将分好组的个体按照 3:1 的比例划分为训练样本集和测试样本集, 使用训练样本集训练改进的集成学习分类器来对子代个体进行分类; 然后, 使用测试样本集计算分类器的误差, 利用两组个体的错误率和可信度格局方法选择有前途的个体进行昂贵适应度评价; 最后, 更新 A_1 和 A_2 归档集, 并对归档集 A_2 进行维护, 剔除劣解, 从而提高代理模型质量. 重复上述过程, 直到满足算法终止条件.

2.1 分类边界选择与分类准则

分类边界选择利用了第1节中基于指标的逐个选择策略从种群 P 中选择一组个体 k , 该策略能够很好地平衡高维目标空间中个体的收敛性与多样性. 由选定的边界个体构造帕累托支配边界, 将归档集 A_2 中的个体分为好解集 (A 组) 和坏解集 (B 组). 分类准则为: 只有不受所有分类边界个体支配的个体才被划分为 A 组; 否则, 将被划分为 B 组. 分类准则如算法2所示.

算法2 分类准则.

输入: 归档集 A_2 , 分类边界 k .

step 1: $A = \emptyset, B = \emptyset$;

step 2: for $i = 1 : |A_2|$;

step 3: $x = A_2(i)$;

step 4: if x 被 k 中的个体支配;

step 5: $B = B \cup x$;

step 6: else

step 7: $A = A \cup x$;

step 8: end if

step 9: end for

step 10: 输出 A 和 B .

在算法2中, 对 A_2 归档集中的个体进行遍历, 将不被所有边界个体支配的个体放入集合 A 中; 否则, 将该个体放入集合 B 中. 最终输出 A 和 B 两个集合.

2.2 改进的集成学习分类模型

本文提出一种改进的装袋集成学习分类器来对子代个体进行分类, 算法流程如算法3所示.

算法3 改进的集成学习分类模型构建.

输入: 模型池大小 T , 被选模型个数 Q , 原始训练样本集 S .

step 1: 计算两两个体间的欧氏距离 d ;

step 2: 根据 d 将原始训练样本集 S 分为 S_i ($i = 1, 2, \dots, T$);

step 3: $S'_i = S - S_i$ ($i = 1, 2, \dots, T$);

step 4: 使用有放回采样技术从 S'_i 中抽取 ϕ_i 样本;

step 5: $D_i = S_i + \phi_i$;

step 6: 使用 D_i 训练 T 个弱分类器;

step 7: 使用模型选择策略从 T 个弱分类器中选择 Q 个;

step 8: 将 Q 个弱分类器的结果进行结合;

step 9: 输出强分类器.

文献[27]指出, 因装袋法中的有放回取样过程具有随机性, 故装袋过程具有盲目性, 尤其当昂贵的适应度评价使得原始训练数据集较小时, 良好的训练样本遍历有助于提升集成学习模型的质量. 改进的集成学习装袋分类器使用文献[27]提供的样本划分方法可以得到 N 个具有多样性和代表性的训练样本子集. 首先, 根据欧氏距离将训练样本划分为 N 个样本子集, 使得每个子集的样本均匀分布在整个样本空间中; 其次, 通过有放回采样方法对每个子集进行采样. 除此之外, 在弱分类器结合之前使用模型选择策略, 从而提升装袋模型的分类质量. 改进的装袋集成学习分类模型如图3所示.

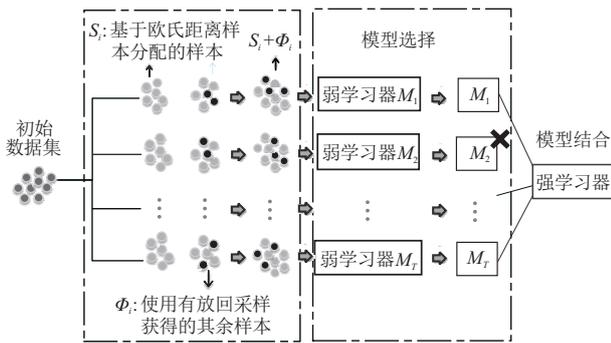


图3 改进装袋集成学习分类模型

改进的装袋集成学习分类模型包括6个部分:基于欧氏距离的样本分配、有放回采样、样本组合、模型训练、模型选择和模型结合。

1) 基于欧氏距离的样本分配. 良好的训练样本集应该均匀分布在整个样本空间中. 样本之间的欧氏距离可以用来衡量样本的分布, 两个样本之间的欧氏距离越大, 样本之间的差异越大. 假设原始训练集为 S , 根据样本间的欧氏距离 d 将 S 划分为 T 个子集 S_1, S_2, \dots, S_T , 每个子集的样本都均匀分布在整个样本空间中.

2) 有放回采样. 通过有放回采样技术从 S'_i 中抽取样本 ϕ_i, S'_i 和 ϕ_i 满足

$$S'_i = S - S_i, \quad i = 1, 2, \dots, T, \quad (8)$$

$$|S_i + \phi_i| = |S|, \quad (9)$$

其中 $|\cdot|$ 代表集合的大小.

3) 样本组合. 基于欧氏距离分配所获得的样本与有放回采样所获得的样本相加即为组合样本, 即

$$D_i = S_i + \phi_i. \quad (10)$$

4) 模型训练. 利用通过样本组合获得的训练子集分别训练 T 个RBF弱分类器.

5) 模型筛选. 通过文献[28]可知, 当集成学习模型规模小于100时, 分类精度随着集成规模的增大而提高. 然而, 当集成规模进一步增大时, 精度却不一定继续提高. 这表明, 使用模型选择策略会有助于在不降低集成分类精度的情况下降低集成规模. 算法根据每个弱分类器的分类均方根误差, 从 T 弱分类器中选择均方根误差最小的 Q 个. 均方根误差计算公式为

$$\text{RMSE} = \sqrt{\sum_i \frac{d_i^2}{i}}, \quad (11)$$

其中 d_i 为预测值与实际值的偏差.

6) 模型结合. 采用平均法对所选的 Q 个弱分类器进行模型结合得到强分类器, 利用强分类器对子代个体进行分类.

2.3 双归档集管理策略

为了增强代理辅助进化算法的收敛性并提高代理模型质量, 算法采用双归档集管理策略, 在每次迭代中更新归档集 A_1 和 A_2 . 对于归档集 A_1 , 将新加入的个体与归档集中原个体进行非支配排序后保留第1层级的个体, 其余个体被舍弃. 对于 A_2 归档集, 如果所有的昂贵适应度个体都用来训练代理模型, 则训练的时长将大大增加, 为了控制代理模型训练的时间, 要对 A_2 归档集进行维护. 将 A_2 内个体数量设置为 N , 当归档集中个体数量大于 N 时, 采用文献[29]中的 k 均值聚类方法对 A_2 归档集进行维护, 通过该方法获得 N 个聚类中心, 将每个类中距聚类中心最近的个体加入 A_2 中.

3 仿真实验及结果分析

3.1 测试问题与性能指标

选取DTLZ^[30]和MaF^[31]两种测试集作为实验中的测试问题, 分别在问题的3、4、6、8、10个目标上测试算法的性能. 设置决策变量为10, 对于DTLZ和MaF上每个测试问题, 每个算法都进行20次独立运算. 采用置信度为0.05的Wilcoxon秩和检验方法进行评判: “+”表示所比较算法的性能优于IECSEA; “-”表示所比较算法的性能不如IECSEA; “=”表示所比较的两种算法之间没有显著差异.

实验对比采用的性能指标是反世代距离(inverted generation distance, IGD), IGD值越小, 表示算法的性能越好. 假设 P^* 是帕累托前沿面上的一组均匀分布的参考点, Ω 是实现的非支配解集. IGD的定义为

$$\text{IGD}(P^*, \Omega) = \frac{\sum_{x \in P^*} \text{dis}(x, \Omega)}{|P^*|}, \quad (12)$$

其中 $\text{dis}(x, \Omega)$ 是 x 与 Ω 中个体的最小欧氏距离.

除了IGD值之外, 实验使用的第2个性能指标是超体积指标(hypervolume, HV). HV值越高, 算法的综合性能越好. HV值定义为给定解集 Ω 的目标空间中一组预定义参考点 P^* 所覆盖的区域, 有

$$\text{HV}(\Omega, P^*) = \lambda(H(\Omega, P^*)). \quad (13)$$

其中

$$H(\Omega, P^*) = \{z \in \mathbf{Z} \mid \exists x \in P, \exists r \in P^* : f(x) \leq z \leq r\}; \quad (14)$$

λ 代表勒贝格测度, 且

$$\lambda(H(\Omega, P^*)) = \int_{P^*} 1_{H(\Omega, P^*)}(z) dz, \quad (15)$$

$1_{H(\Omega, P^*)}$ 为 $H(\Omega, P^*)$ 的特征函数.

3.2 对比算法

实验选择 4 种算法与 IECSEA 进行对比,所选的 4 种对比算法简要介绍如下.

1) NSGA-III^[32]: NSGA-III 除了使用 NSGA-II 的框架之外,还使用了一组预先定义的参考点. 虽然 NSGA-III 不是一个代理辅助进化算法,但是在本实验中它也被用来与代理辅助进化算法对比,用于体现在有限的评价次数下, SAEAs 与传统进化算法相比其优势所在.

2) MOEA/D-EGO^[14]: MOEA/D-EGO 将多目标优化问题分解为多个单目标的子问题,再为每个子问题建立一个代理模型.

3) CPS-MOEA^[15]: CPS-MOEA 是一种基于分类的代理辅助进化算法. 它使用两个外部归档集训练分类器,然后使用分类回归树过滤新生成的候选个体,只保留正解.

4) CSEA^[16]: CSEA 也是一种基于分类的代理辅助进化算法,它使用 FNN 作为代理模型来预测候选个体与参考个体之间的支配关系. 利用预测中的不确定性信息选择有前途的个体,并用昂贵适应度函数进行评价.

3.3 实验参数设定

为了对所有对比算法进行公平比较并获得最佳性能,对比算法都使用其原文所推荐的参数设置. 具体情况如下:

1) 遗传操作: 交叉分布指数 n_c 设为 20, 变异分布指数 n_m 设为 20, 交叉概率 p_c 设为 1.0, 变异概率 p_m 设为 $1/d$, 其中 d 为决策变量个数.

2) 种群规模: 为便于比较,实验中将 CPS-MOEA、CSEA 和 IECSEA 的种群规模设置为 $11d-1$, d 为决策变量个数. 对于 NSGA-III 和 MOEA/D-EGO, 表 1 列出了两个参数 (p_1, p_2) 的设置, 用于为不同的目标生成最接近 $11d-1$ 的总体规模, 其中 p_1 和 p_2 是控制沿着帕累托前沿边界及其内部的权重点数量的参数.

表 1 NSGA-III 和 MOEA/D-EGO 中最接近 $11d-1$ 种群规模的参数设置量

obj.	parameter(p_1, p_2)	population size(N)
3	(12, 0)	105
4	(6, 0)	84
6	(3, 2)	77
8	(2, 2)	72
10	(2, 1)	65

3) 实验终止条件: 对于所有测试问题, 终止条件为 FEs 的最大数目. 由于所涉及的问题是昂贵的, FEs

的最大数目被设置为 400.

4) 各算法特有参数设置: 对于 CSEA, 参考解的个数设为 6 个, FNN 中隐藏神经元的个数设为 10 个. 对于 IECSEA, 分类边界个体数量设置为 8, 集成学习模型池的大小为 $T = 2000$, 被选择的模型数为 $Q = 100$.

3.4 实验结果与分析

本节首先验证所提出的改进集成学习分类代理模型的有效性; 其次将算法与 NSGA-III、CPS-MOEA、MOEA/D-EGO 和 CSEA 算法在 DTLZ 和 MaF 系列测试集上 IGD 和 HV 指标的表现进行对比, 用于判断这几种算法的综合性能.

3.4.1 代理模型有效性验证

为了验证基于改进集成学习分类的代理模型的有效性, 本节在算法其他流程不变的条件下, 将使用改进集成学习分类器作为代理模型的 IECSEA 算法与使用 CART、FNN、SVM 及 RBF 作为代理模型的 IECSEA-C、IECSEA-F、IECSEA-S、IECSEA-R 算法在 3、4、6、8、10 目标的 DTLZ 问题上运行的 IGD 值进行对比. 统计结果如表 2 所示, 其中突出显示了最佳结果. 图 4 绘制了这 5 种算法在 3、8 目标的 DTLZ2 问题上运行的 IGD 值变化曲线.

由实验结果可知: 当代理模型为改进集成学习分类器时, 算法表现出了最好的结果; 当代理模型为 CART 和 SVM 时, 结果表现最差. 由此可以说明, 当基于分类的代理辅助进化算法用于解决昂贵高维多目标优化问题时, 算法的性能与所选择的代理模型有关, 本文所提出的基于改进集成学习分类的代理模型可以有效提升基于分类的代理辅助进化算法求解昂贵高维多目标优化问题的性能.

3.4.2 DTLZ 测试问题结果与分析

表 3 和表 4 显示了 5 种算法在 DTLZ1 ~ DTLZ5 测试问题上进行 20 次独立运行所获得的 IGD 和 HV 值的统计结果. 由表 3 和表 4 可知: IECSEA 算法在 DTLZ2、DTLZ4 和 DTLZ5 测试问题上获得了最好的效果; 虽然在 DTLZ3 问题上稍逊色于 CSEA 算法, 但总体上显示出了算法的竞争力.

当解决 DTLZ1 和 DTLZ3 这类具有多模态的问题时, 主要难点在于要用少量的真实评价来获得一组收敛性好的个体. 在这类问题上, IECSEA 算法和 CSEA 算法的结果表现最好, 其次是 CPS-MOEA 算法, 这说明基于分类的代理辅助进化算法适合求解这类具有多个局部最优的多模态的问题, 因为其分类器可用于探索未知空间. 图 5(a) 绘制了所对比的 4 个代

表2 使用不同代理模型时算法的IGD性能指标

problem	M	IECSEA-C	IECSEA-F	IECSEA-S	IECSEA-R	IECSEA
DTLZ1	3	6.3873e+1 (1.90e+1) -	6.0876e+1 (1.54e+1) =	6.3123e+1 (1.94e+1) -	6.0704e+1 (1.20e+1) =	5.8373e+1 (1.02e+1)
	4	4.3549e+1 (1.29e+1) -	3.8106e+1 (1.44e+1) =	3.7948e+1 (1.47e+1) =	4.2057e+1 (1.03e+1) -	3.6772e+1 (9.11e+0)
	6	3.4652e+1 (6.51e+0) -	2.1002e+1 (6.21e+0) =	3.0832e+1 (4.91e+0) -	1.8974e+1 (5.28e+0) =	2.0415e+1 (5.41e+0)
	8	3.6598e+0 (2.04e+0) =	4.0338e+0 (1.15e+0) -	4.5722e+0 (2.33e+0) -	3.9449e+0 (2.02e+0) -	3.6158e+0 (2.39e+0)
	10	2.3396e-1 (6.89e-2) +	2.8283e-1 (3.85e-2) =	2.5361e-1 (7.82e-2) +	2.9263e-1 (1.00e-1) =	2.9608e-1 (3.56e-2)
DTLZ2	3	2.3701e-1 (2.31e-2) -	1.8205e-1 (2.58e-2) -	2.3680e-1 (2.76e-2) -	2.0910e-1 (2.50e-2) -	1.5143e-1 (2.04e-2)
	4	2.5575e-1 (2.13e-2) -	2.5784e-1 (1.78e-2) -	2.7959e-1 (3.34e-2) -	2.7973e-1 (2.05e-2) -	2.0672e-1 (1.27e-2)
	6	3.6772e-1 (3.06e-2) =	4.0496e-1 (2.25e-2) =	3.9775e-1 (3.28e-2) =	4.0537e-1 (3.28e-2) =	3.7611e-1 (4.70e-2)
	8	6.6389e-1 (4.62e-2) -	5.9249e-1 (2.13e-2) -	6.0624e-1 (3.89e-2) -	6.1924e-1 (2.98e-2) -	5.2442e-1 (4.45e-2)
	10	5.9540e-1 (4.96e-2) =	6.0271e-1 (6.16e-2) =	6.5239e-1 (4.15e-2) -	5.4506e-1 (3.80e-2) +	5.8578e-1 (4.33e-2)
DTLZ3	3	1.5520e+2 (5.12e+1) -	1.3907e+2 (1.80e+1) =	1.9546e+2 (3.18e+1) -	1.6229e+2 (5.41e+1) -	1.1917e+2 (4.79e+1)
	4	3.0533e+2 (2.97e+1) -	2.0081e+2 (3.30e+1) -	1.3315e+2 (3.01e+1) =	1.1770e+2 (2.76e+1) =	1.2214e+2 (1.49e+1)
	6	6.2639e+1 (1.07e+1) =	5.1536e+1 (1.45e+1) +	6.9787e+1 (1.87e+1) -	5.9666e+1 (1.37e+1) +	6.4963e+1 (1.88e+1)
	8	9.3641e+0 (5.40e+0) +	7.0775e+0 (6.25e+0) +	1.2650e+1 (7.10e+0) +	1.9409e+1 (1.05e+1) =	1.8834e+1 (1.86e+1)
	10	7.7208e-1 (2.81e-2) =	8.2191e-1 (2.51e-2) -	1.1001e+0 (3.19e-1) -	9.0895e-1 (1.95e-1) -	7.9233e-1 (3.16e-1)
DTLZ4	3	4.5173e-1 (1.04e-1) -	4.4464e-1 (1.96e-1) -	3.6028e-1 (9.61e-2) -	4.3480e-1 (1.27e-1) -	2.8017e-1 (1.64e-2)
	4	4.6335e-1 (8.62e-2) -	3.8259e-1 (1.04e-1) -	3.9364e-1 (4.98e-2) -	3.7845e-1 (9.06e-2) -	3.2837e-1 (1.09e-2)
	6	6.2699e-1 (2.99e-2) -	5.1754e-1 (6.95e-2) =	6.3179e-1 (5.42e-2) -	5.5741e-1 (4.21e-2) =	5.3999e-1 (2.21e-2)
	8	6.6204e-1 (5.49e-2) -	5.9859e-1 (7.02e-2) -	6.7720e-1 (4.02e-2) -	5.4761e-1 (4.56e-2) =	5.4398e-1 (8.50e-2)
	10	6.9284e-1 (3.43e-2) =	6.1823e-1 (2.01e-2) +	7.3160e-1 (2.74e-2) -	6.5277e-1 (2.96e-2) =	6.7181e-1 (2.09e-3)
DTLZ5	3	7.8395e-2 (2.75e-2) =	7.5644e-2 (2.40e-2) +	1.0033e-1 (2.48e-2) -	9.5797e-2 (2.62e-2) -	8.0071e-2 (1.07e-2)
	4	2.7663e-1 (2.53e-2) -	1.8534e-1 (2.16e-2) -	1.7578e-1 (1.45e-2) -	1.8235e-1 (1.92e-2) -	1.1315e-1 (2.49e-2)
	6	5.2918e-2 (2.27e-2) -	4.5057e-2 (1.14e-2) =	5.1209e-2 (1.75e-2) -	5.2404e-2 (1.02e-2) -	4.4563e-2 (4.97e-3)
	8	3.9815e-2 (9.78e-3) -	3.0876e-2 (4.37e-3) -	2.6784e-2 (6.72e-3) =	3.0276e-2 (1.01e-2) -	2.6024e-2 (7.09e-3)
	10	1.5415e-2 (1.56e-3) -	1.2756e-2 (1.50e-3) =	2.1047e-2 (9.14e-4) -	1.3901e-2 (9.45e-4) =	1.1332e-2 (1.48e-3)
+/-/=		2/16/7	4/11/10	2/19/4	2/13/10	

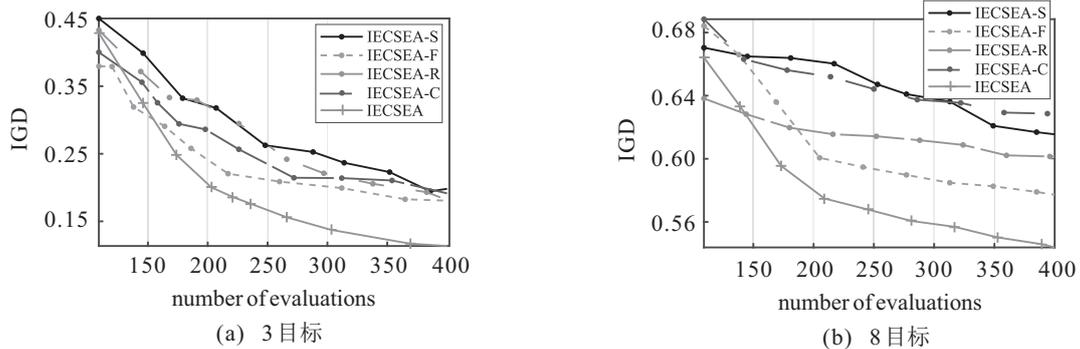


图4 5种算法在3、8目标的DTLZ2问题上的IGD值变化曲线

理辅助进化算法在3个目标的DTLZ1测试问题上的IGD中值所对应的非支配解。由图5可知,所提出的IECSEA算法通过提高代理模型的分类型精度能有效地加快算法收敛速度。从表4的HV统计结果可以发现,5种对比算法在DTLZ1和DTLZ3测试问题上的HV值大多为0,说明这5种算法得到的非支配解集离真实的帕累托前沿还很远,需要更多次的函数评价才能解决这类问题。

DTLZ2和DTLZ4测试问题相较于其他测试问题是容易收敛的,但要获得均匀分布在整个帕累托前沿面上的解集则很难保持种群的多样性。它们通常用于评估算法是否能保持良好的个体分布性,其中DTLZ2测试问题是具有凹面椭圆的问题,IECSEA算法在DTLZ2测试问题上综合性能表现最好。图5(b)可视化了4种代理辅助进化算法在具有8个目标的DTLZ2测试问题上的平行坐标图,由该图可

表 3 各种对比算法在 DTLZ 问题上的 IGD 性能指标

problem	M	NSGA-III	CPS-MOEA	MOEA/D-EGO	CSEA	IECSEA
DTLZ 1	3	6.803 9e+1 (1.11e+1) -	6.679 0e+1 (8.89e+0) -	8.777 8e+1 (1.43e+1) -	5.015 0e+1 (2.12e+1) +	5.837 3e+1 (1.02e+1)
	4	5.331 4e+1 (1.41e+1) -	5.180 8e+1 (1.07e+1) -	5.297 9e+1 (5.48e+0) -	3.962 0e+1 (6.85e+0) -	3.677 2e+1 (9.11e+0)
	6	2.724 4e+1 (5.45e+0) -	2.718 0e+1 (2.13e+0) -	2.969 6e+1 (8.63e+0) -	2.124 3e+1 (9.16e-1) =	2.041 5e+1 (5.41e+0)
	8	8.855 0e+0 (1.71e+0) -	9.409 2e+0 (2.77e+0) -	1.170 2e+1 (2.17e+0) -	3.906 6e+0 (2.65e+0) -	3.615 8e+0 (2.39e+0)
	10	3.973 5e-1 (9.20e-2) -	3.478 1e-1 (5.08e-2) -	3.679 4e-1 (9.68e-2) -	2.769 5e-1 (1.22e-2) =	2.960 8e-1 (3.56e-2)
DTLZ 2	3	2.640 4e-1 (1.84e-2) -	2.744 2e-1 (3.54e-2) -	3.427 0e-1 (3.73e-2) -	1.891 4e-1 (2.11e-2) -	1.514 3e-1 (2.04e-2)
	4	3.253 4e-1 (2.46e-2) -	3.693 9e-1 (1.41e-2) -	3.729 1e-1 (1.53e-2) -	2.825 7e-1 (2.85e-2) -	2.067 2e-1 (1.27e-2)
	6	4.431 0e-1 (5.18e-2) -	5.266 9e-1 (1.56e-2) -	4.488 7e-1 (2.48e-2) -	4.413 5e-1 (6.51e-2) -	3.761 1e-1 (4.70e-2)
	8	6.134 1e-1 (4.42e-2) -	5.927 6e-1 (1.55e-2) -	5.001 7e-1 (1.92e-2) =	6.226 9e-1 (2.39e-2) -	5.244 2e-1 (4.45e-2)
	10	6.299 4e-1 (4.39e-2) -	6.159 8e-1 (1.89e-2) =	5.004 8e-1 (9.40e-3) +	6.983 4e-1 (1.90e-2) -	5.857 8e-1 (4.33e-2)
DTLZ 3	3	2.585 2e+2 (2.81e+1) -	1.617 1e+2 (4.26e+1) -	1.913 0e+2 (1.01e+1) -	1.020 6e+2 (2.02e+1) +	1.191 7e+2 (4.79e+1)
	4	1.634 5e+2 (6.59e+1) -	1.451 9e+2 (2.63e+1) -	1.684 6e+2 (6.66e+0) -	8.971 2e+1 (1.07e+1) +	1.221 4e+2 (1.49e+1)
	6	6.970 4e+1 (3.11e+1) =	8.346 8e+1 (2.10e+1) -	8.812 4e+1 (1.18e+1) -	4.829 9e+1 (1.02e+1) +	6.496 3e+1 (1.88e+1)
	8	2.759 4e+1 (1.01e+1) -	4.072 9e+1 (9.42e+0) -	3.242 5e+1 (1.05e+1) -	6.874 5e+0 (4.10e+0) +	1.883 4e+1 (1.86e+1)
	10	1.529 2e+0 (5.23e-1) -	3.832 7e+0 (4.47e+0) -	1.144 0e+0 (1.63e-1) -	1.001 3e+0 (2.57e-1) -	7.923 3e-1 (3.16e-1)
DTLZ 4	3	5.849 8e-1 (3.54e-2) -	5.336 9e-1 (2.94e-2) -	5.987 1e-1 (7.81e-2) -	3.642 6e-1 (1.35e-1) -	2.801 7e-1 (1.64e-2)
	4	6.557 6e-1 (7.46e-2) -	5.964 2e-1 (2.46e-2) -	6.478 1e-1 (3.52e-2) -	3.975 7e-1 (1.31e-1) -	3.283 7e-1 (1.09e-2)
	6	6.865 8e-1 (7.45e-2) -	6.302 9e-1 (1.02e-2) -	6.880 1e-1 (2.06e-2) -	5.207 5e-1 (1.21e-1) =	5.399 9e-1 (2.21e-2)
	8	7.546 6e-1 (9.33e-2) -	6.190 9e-1 (1.37e-2) -	6.385 1e-1 (1.06e-2) -	5.572 5e-1 (3.20e-2) =	5.439 8e-1 (8.50e-2)
	10	7.514 0e-1 (6.59e-2) -	6.435 5e-1 (5.62e-3) =	6.266 2e-1 (1.87e-2) +	6.356 3e-1 (3.16e-2) +	6.718 1e-1 (2.09e-3)
DTLZ 5	3	2.041 4e-1 (2.91e-2) -	1.976 0e-1 (6.81e-3) -	2.609 6e-1 (3.94e-2) -	7.704 1e-2 (2.84e-2) =	8.007 1e-2 (1.07e-2)
	4	1.773 7e-1 (3.66e-2) -	1.701 8e-1 (3.01e-2) -	1.921 3e-1 (3.38e-2) -	1.863 6e-1 (2.44e-2) -	1.131 5e-1 (2.49e-2)
	6	1.176 4e-1 (1.89e-2) -	1.655 9e-1 (3.65e-2) -	1.611 9e-1 (3.33e-2) -	4.294 4e-2 (1.13e-2) =	4.456 3e-2 (4.97e-3)
	8	8.376 0e-2 (1.94e-2) -	9.197 8e-2 (2.22e-2) -	7.731 5e-2 (1.00e-2) -	3.303 0e-2 (4.74e-3) -	2.602 4e-2 (7.09e-3)
	10	6.428 8e-2 (1.85e-2) -	3.526 0e-2 (6.05e-3) -	2.043 9e-2 (2.03e-3) -	1.279 2e-2 (1.81e-3) =	1.133 2e-2 (1.48e-3)
+/-/=	0/24/1	0/23/2	2/22/1	6/12/7		

表 4 各种对比算法在 DTLZ 问题上的 HV 性能指标

problem	M	NSGA-III	CPS-MOEA	MOEA/D-EGO	CSEA	IECSEA
DTLZ 1	3	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0)
	4	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0)
	6	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0)
	8	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0)
	10	2.927 3e-1 (2.83e-1) -	3.531 3e-1 (1.20e-1) -	2.839 1e-1 (1.84e-1) -	7.600 5e-1 (1.21e-1) +	5.889 2e-1 (1.39e-1)
DTLZ 2	3	1.889 5e-1 (2.04e-2) -	1.939 6e-1 (3.74e-2) -	1.398 9e-1 (6.36e-2) -	3.640 6e-1 (5.25e-2) -	4.098 2e-1 (2.62e-2)
	4	2.531 0e-1 (3.78e-2) -	2.005 4e-1 (3.11e-2) -	2.199 0e-1 (2.54e-2) -	3.964 0e-1 (6.26e-2) -	5.289 5e-1 (3.19e-2)
	6	3.711 0e-1 (4.76e-2) -	2.064 8e-1 (2.33e-2) -	4.133 1e-1 (4.26e-2) -	4.930 3e-1 (1.01e-1) -	6.081 9e-1 (4.95e-2)
	8	4.212 8e-1 (1.84e-2) -	3.586 8e-1 (3.37e-2) -	6.091 1e-1 (2.88e-2) -	5.169 5e-1 (3.08e-2) -	6.647 8e-1 (3.85e-2)
	10	5.805 8e-1 (2.17e-2) -	5.652 6e-1 (3.54e-2) -	8.534 6e-1 (1.04e-2) +	5.930 6e-1 (4.65e-2) -	7.332 4e-1 (2.54e-2)
DTLZ 3	3	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0)
	4	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0)
	6	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0)
	8	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0) =	0.000 0e+0 (0.00e+0)
	10	8.351 0e-2 (1.16e-1) -	1.185 5e-2 (2.12e-2) -	6.592 2e-2 (7.75e-2) -	3.133 2e-1 (1.97e-1) =	2.835 3e-1 (1.00e-1)
DTLZ 4	3	1.505 4e-2 (1.80e-2) -	2.122 0e-2 (1.69e-2) -	2.524 0e-2 (4.60e-2) -	2.529 3e-1 (3.16e-3) -	3.174 9e-1 (8.12e-2)
	4	9.859 6e-2 (4.99e-2) -	5.298 1e-2 (2.14e-2) -	1.067 0e-2 (1.14e-2) -	3.005 0e-1 (2.45e-2) -	4.793 9e-1 (8.62e-2)
	6	2.288 0e-1 (6.24e-2) -	1.456 6e-1 (4.58e-2) -	8.113 8e-2 (2.52e-2) -	5.783 1e-1 (8.07e-2) =	5.759 0e-1 (1.80e-2)
	8	2.967 0e-1 (6.17e-2) -	4.658 9e-1 (6.88e-2) -	3.389 8e-1 (6.97e-2) -	7.477 6e-1 (2.13e-2) +	7.005 1e-1 (7.40e-2)
	10	6.140 3e-1 (8.08e-2) -	7.559 3e-1 (1.82e-2) =	8.138 8e-1 (4.20e-2) +	7.767 5e-1 (3.62e-2) =	7.756 4e-1 (2.57e-3)
DTLZ 5	3	2.920 0e-2 (7.13e-3) -	3.261 1e-2 (9.90e-3) -	2.133 8e-2 (1.59e-2) -	1.410 1e-1 (1.46e-2) =	1.219 1e-1 (1.25e-2)
	4	2.001 4e-2 (8.93e-3) -	3.001 7e-2 (2.37e-2) -	5.510 7e-2 (2.64e-2) -	7.843 9e-2 (1.41e-2) -	9.648 4e-2 (1.96e-2)
	6	4.976 4e-2 (2.98e-2) -	4.307 6e-2 (1.34e-2) -	5.104 6e-2 (2.87e-2) -	9.735 6e-2 (4.97e-3) +	8.931 7e-2 (1.43e-2)
	8	6.066 3e-2 (1.34e-2) -	6.918 7e-2 (1.79e-2) -	8.655 2e-2 (6.30e-3) -	9.909 3e-2 (2.00e-3) =	9.933 2e-2 (4.36e-3)
	10	8.508 3e-2 (7.90e-3) -	9.862 9e-2 (1.35e-4) =	9.700 4e-2 (5.49e-4) -	9.536 1e-2 (4.55e-4) -	1.004 6e-1 (5.12e-4)
+/-/=	0/17/8	0/15/10	2/15/8	3/9/13		

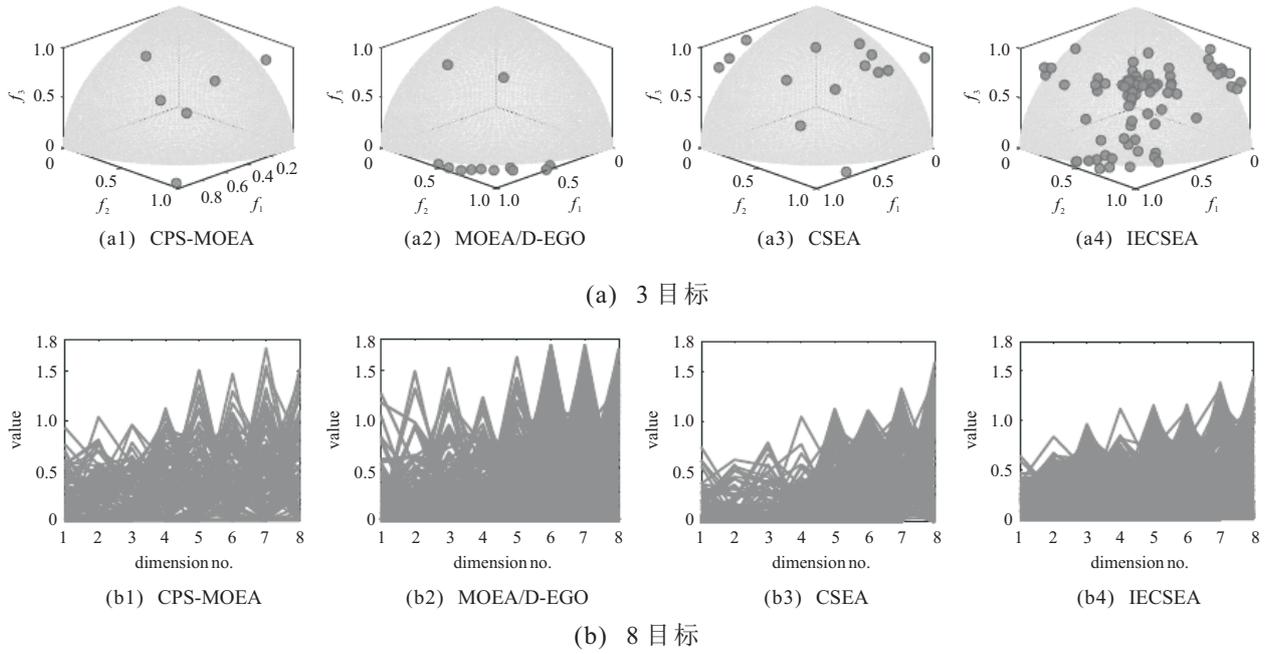


图5 4种SAEAs在3目标DTLZ1问题上的IGD中值所对应的非支配解及在8目标DTLZ2问题上的平行坐标图

表5 各种对比算法在MaF问题上的IGD性能指标

problem	M	NSGA-III	CPS-MOEA	MOEA/D-EGO	CSEA	IECSEA
MaF1	3	2.700 3e-1 (3.12e-3) -	2.672 9e-1 (1.62e-2) -	2.529 7e-1 (1.15e-2) -	1.458 6e-1 (3.64e-2) -	1.032 2e-1 (1.60e-2)
	4	3.041 7e-1 (7.14e-2) -	3.191 4e-1 (1.11e-2) -	3.359 3e-1 (2.09e-2) -	1.898 3e-1 (2.94e-3) -	1.558 5e-1 (1.92e-2)
	6	3.488 2e-1 (2.05e-3) -	3.344 3e-1 (1.93e-2) -	4.072 1e-1 (2.96e-2) -	2.747 5e-1 (5.79e-2) =	2.632 0e-1 (2.30e-2)
	8	3.907 5e-1 (2.00e-2) -	3.975 6e-1 (4.94e-3) -	4.231 4e-1 (3.45e-2) -	2.364 0e-1 (1.46e-2) +	3.369 0e-1 (3.68e-2)
	10	4.080 7e-1 (2.41e-2) -	3.467 2e-1 (3.56e-2) -	3.489 0e-1 (5.40e-3) -	2.537 1e-1 (7.38e-3) =	2.581 4e-1 (8.87e-3)
MaF2	3	5.661 6e-2 (2.61e-3) -	6.016 5e-2 (3.20e-3) -	5.366 2e-2 (3.02e-3) -	5.090 1e-2 (2.83e-3) -	4.261 3e-2 (8.39e-4)
	4	9.514 7e-2 (1.46e-3) -	9.764 0e-2 (5.78e-4) -	7.802 4e-2 (9.52e-4) -	7.693 8e-2 (2.30e-4) -	6.686 9e-2 (4.88e-4)
	6	2.030 6e-1 (1.29e-2) =	1.494 1e-1 (2.22e-3) +	1.848 0e-1 (2.09e-2) =	2.139 2e-1 (7.08e-3) =	2.079 5e-1 (1.23e-2)
	8	2.943 7e-1 (7.09e-3) =	2.275 3e-1 (1.10e-2) +	2.698 9e-1 (1.12e-2) =	2.658 7e-1 (4.71e-3) =	2.818 3e-1 (4.89e-3)
	10	2.865 5e-1 (3.68e-2) +	2.347 0e-1 (2.51e-2) +	2.986 4e-1 (7.51e-3) +	3.221 0e-1 (3.25e-3) =	3.398 0e-1 (2.50e-4)
MaF3	3	2.468 6e+5 (2.59e+5) -	7.105 6e+4 (4.23e+4) -	9.882 0e+4 (1.69e+4) -	1.459 6e+4 (7.14e+3) +	4.622 8e+4 (3.31e+4)
	4	1.265 2e+5 (1.07e+5) -	5.918 8e+4 (3.77e+4) -	1.972 5e+5 (4.41e+4) -	4.856 1e+3 (3.47e+4) +	2.552 0e+4 (6.76e+4)
	6	9.739 1e+4 (5.71e+4) -	5.153 4e+4 (2.07e+4) -	4.932 2e+4 (4.92e+4) -	4.995 9e+3 (2.84e+3) -	3.135 2e+3 (1.50e+4)
	8	7.493 1e+2 (5.05e+3) +	5.858 3e+3 (6.01e+3) -	3.135 4e+3 (5.75e+3) -	6.737 8e+2 (2.14e+3) +	1.286 6e+3 (1.31e+2)
	10	2.824 7e-1 (1.64e+0) +	2.870 3e+1 (4.10e+1) -	5.239 7e-1 (2.05e+1) +	2.086 1e-1 (1.79e-1) +	9.899 9e-1 (2.48e-1)
MaF4	3	9.614 5e+2 (2.09e+2) -	8.105 5e+2 (1.33e+2) -	6.163 0e+2 (1.57e+2) -	7.124 2e+2 (6.74e+1) -	2.786 4e+2 (1.83e+1)
	4	1.871 9e+3 (2.71e+2) -	1.784 0e+3 (2.29e+2) -	1.690 4e+3 (3.14e+2) -	8.440 2e+2 (2.53e+2) -	3.983 6e+2 (1.25e+2)
	6	2.987 5e+3 (1.97e+3) -	2.264 0e+3 (3.06e+2) =	3.398 1e+3 (3.22e+2) -	1.070 3e+3 (2.85e+2) +	2.426 1e+3 (2.85e+2)
	8	3.269 9e+3 (1.15e+3) -	3.843 1e+3 (1.05e+3) -	4.498 4e+3 (2.21e+3) -	8.568 3e+2 (2.54e+2) +	1.499 8e+3 (3.24e+2)
	10	7.777 9e+2 (1.29e+3) -	3.585 7e+2 (2.09e+2) -	1.735 3e+2 (1.62e+3) +	2.441 5e+2 (3.86e+1) +	2.946 9e+2 (7.58e+1)
MaF5	3	2.383 7e+0 (7.25e-1) -	2.227 0e+0 (6.78e-1) -	2.433 8e+0 (4.25e-1) -	2.245 8e+0 (6.23e-1) -	1.331 6e+0 (3.68e+0)
	4	4.218 8e+0 (5.15e-1) -	4.261 2e+0 (4.04e-1) -	4.766 8e+0 (3.49e-1) -	2.155 4e+0 (5.12e-1) +	2.754 4e+0 (2.13e+0)
	6	1.286 6e+1 (8.36e-1) -	1.292 3e+1 (5.20e-1) -	1.203 8e+1 (7.43e-1) -	9.452 0e+0 (1.73e+0) -	6.684 8e+0 (1.99e+0)
	8	3.454 5e+1 (4.20e+0) -	3.683 1e+1 (2.35e+0) -	2.958 4e+1 (4.26e+0) -	3.738 3e+1 (6.15e-1) -	2.143 1e+1 (2.57e+0)
	10	1.220 5e+2 (1.58e+1) =	1.152 7e+2 (7.79e+0) =	9.811 8e+1 (3.92e+1) +	1.060 2e+2 (2.70e+1) =	1.053 3e+2 (2.46e+1)
+/-/=		3/19/3	3/20/2	4/19/2	9/10/6	

知, IECSEA 算法得到的解的范围比其他 3 种算法得到的解的范围要大, 即 IECSEA 算法的解具有更好的分布性. 与 DTLZ2 测试问题相比, DTLZ4 测试问题的难点在于它的帕累托前沿上的点具有很强的偏好, 在 DTLZ4 测试问题中, IECSEA 算法表现最好, 其次是 CSEA 算法, 这是因为基于分类的代理辅助进化算法可以快速定位偏好区域. 此外还可以发现, MOEA/D-EGO 算法在 8、10 目标的 DTLZ2 和 DTLZ4 测试问题上表现良好, 这是因为 MOEA/D-EGO 算法采用了基于聚类的选择策略, 有助于算法在高维目标空间中个体的分布性保持.

DTLZ5 测试问题因具有退化的特点而导致其很难收敛, IECSEA 算法能够很好地处理这类退化问题. 虽然 IECSEA 算法并不是在所有的 DTLZ 测试问题上都表现出最好的结果, 但与其他 4 种算法相比, 已经表现出了很强的竞争力.

3.4.3 MaF 测试集结果与分析

表 5 展示了 5 种算法在 MaF 测试问题上的 IGD 值. 由表 5 的实验结果可知, 在 MaF 1 和 MaF 5 问题上, IECSEA 算法表现出了最佳的结果, 其次是 CSEA 算法. IECSEA 算法在 3、4 目标的 MaF 2 问题上表现最佳, CPS-MOEA 算法在 6、8、10 目标的 MaF 2 问题上表现最佳, 虽然 IECSEA 算法在高目标的 MaF 2 上的 IGD 值不如 CPS-MOEA 算法, 但它已经表现出比 CSEA 算法更好的结果. 在 MaF 3 问题上 CSEA 算法表现最优, 其次是 IECSEA 算法, 说明 CSEA 算法更适合求解这类问题. 在 MaF 4 问题上 IECSEA 算法与 CSEA 算法的表现相似. 由 MaF 问题上的实验结果可知, IECSEA 算法和 CSEA 算法在整体性能上的表现都优于其他算法, 这也进一步验证了基于分类的代理辅助进化算法在求解这类问题时的优势.

4 结论

本文提出了一种基于改进集成学习分类的代理辅助进化算法 IECSEA, 用于解决昂贵高维多目标优化问题. 算法采用逐个选择策略选择一组分类边界个体来将候选个体分为两组, 使用了一种改进的集成学习分类器作为代理模型来预测子代个体的类别, 再通过可信度格局选择有前途的个体用于昂贵的适应度评价, 并且通过双归档集管理策略管理和维护 A_1 和 A_2 两个归档集. 实验通过对比不同的代理模型验证了所提出的改进集成学习分类器的优越性, 并将 IECSEA 在广泛使用的 DTLZ 和 MaF 测试问题上与 NSGA-III、CPS-MOEA、MOEA/D-EGO 和 CSEA 进行了比较, 验证了其有效性, 也进一步表明了基于分类

的 SAEAs 在解决昂贵高维多目标优化问题上的竞争力.

参考文献(References)

- [1] 刘建昌, 李飞, 王洪海, 等. 进化高维多目标优化算法研究综述[J]. 控制与决策, 2018, 33(5): 879-887.
(Liu J C, Li F, Wang H H, et al. Survey on evolutionary many-objective optimization algorithms[J]. Control and Decision, 2018, 33(5): 879-887.)
- [2] Gu Q H, Chen H Y, Chen L, et al. A many-objective evolutionary algorithm with reference points-based strengthened dominance relation[J]. Information Sciences, 2021, 554: 236-255.
- [3] Jin Y C, Sendhoff B. A systems approach to evolutionary multiobjective 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(S2): W615-W621.
- [5] Jin Y C. Surrogate-assisted evolutionary computation: Recent advances and future challenges[J]. Swarm and Evolutionary Computation, 2011, 1(2): 61-70.
- [6] Gu Q H, Wang Q, Li X X, et al. A surrogate-assisted multi-objective particle swarm optimization of expensive constrained combinatorial optimization problems[J]. Knowledge-Based Systems, 2021, 223: 107049.
- [7] 顾清华, 周煜丰, 李学现. 基于径向空间划分的昂贵多目标进化算法[J]. 自动化学报, 2021, 47: 1-21.
(Gu Q H, Zhou Y F, Li X X. Expensive multi-objective evolutionary algorithm based on radial space partition[J]. Acta Automatica Sinica, 2021, 47: 1-21.)
- [8] Wilson B, Cappelleri D, Simpson T W. Efficient Pareto frontier exploration using surrogate approximations[J]. Optimization and Engineering, 2001, 2(1): 31-50.
- [9] Knowles J. ParEGO: A hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems[J]. IEEE Transactions on Evolutionary Computation, 2006, 10(1): 50-66.
- [10] Theil H. A rank-Invariant method of linear and polynomial regression analysis[C]. Henri Theil's Contributions to Economics and Econometrics. Dordrecht: Springer, 1992: 345-381.
- [11] Jeong S, Murayama M, Yamamoto K. Efficient optimization design method using kriging model[J]. Journal of Aircraft, 2005, 42(2): 413-420.
- [12] Dreiseitl S, Ohno-Machado L. Logistic regression and artificial neural network classification models: A methodology review[J]. Journal of Biomedical

- Informatics, 2002, 35(5/6): 352-359.
- [13] Jin Y C, Olhofer M, Sendhoff B. A framework for evolutionary optimization with approximate fitness functions[J]. IEEE Transactions on Evolutionary Computation, 2002, 6(5): 481-494.
- [14] Zhang Q F, Liu W D, Tsang E, et al. Expensive multiobjective optimization by MOEA/D with Gaussian process model[J]. IEEE Transactions on Evolutionary Computation, 2010, 14(3): 456-474.
- [15] Zhang J Y, Zhou A M, Zhang G X. A classification and Pareto domination based multiobjective evolutionary algorithm[C]. IEEE Congress on Evolutionary Computation. Sendai, 2015: 2883-2890.
- [16] Pan L Q, He C, Tian Y, et al. A classification-based surrogate-assisted evolutionary algorithm for expensive many-objective optimization[J]. IEEE Transactions on Evolutionary Computation, 2019, 23(1): 74-88.
- [17] Cortes C, Vapnik V. Support-vector networks[J]. Machine Learning, 1995, 20(3): 273-297.
- [18] Er M J, Wu S Q, Lu J W, et al. Face recognition with radial basis function (RBF) neural networks[J]. IEEE Transactions on Neural Networks, 2002, 13(3): 697-710.
- [19] Ren Y, Zhang L, Suganthan P N. Ensemble classification and regression-recent developments, applications and future directions[J]. IEEE Computational Intelligence Magazine, 2016, 11(1): 41-53.
- [20] Seni G, Elder J F. Ensemble methods in data mining: Improving accuracy through combining predictions[J]. Synthesis Lectures on Data Mining and Knowledge Discovery, 2010, 2(1): 1-126.
- [21] Ji X F, Zhang Y, Gong D W, et al. Dual-surrogate-assisted cooperative particle swarm optimization for expensive multimodal problems[J]. IEEE Transactions on Evolutionary Computation, 2021, 25(4): 794-808.
- [22] Sagi O, Rokach L. Ensemble learning: A survey[J]. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 2018, 8(4): e1249.
- [23] Schapire R E. The boosting approach to machine learning: An overview[J]. Nonlinear Estimation and Classification, 2003: 149-171.
- [24] Breiman L. Bagging predictors[J]. Machine Learning, 1996, 24(2): 123-140.
- [25] Liu Y P, Gong D W, Sun J, et al. A many-objective evolutionary algorithm using a one-by-one selection strategy[J]. IEEE Transactions on Cybernetics, 2017, 47(9): 2689-2702.
- [26] Beasley D, Bull D R, Martin R R. A sequential niche technique for multimodal function optimization[J]. Evolutionary Computation, 1993, 1(2): 101-125.
- [27] Chen R Q, Yu J S. An improved bagging neural network ensemble algorithm and its application[C]. The 3rd International Conference on Natural Computation (ICNC 2007). Haikou, 2007: 730-734.
- [28] Zhou Z H. Ensemble methods: Foundations and algorithms[M]. Boca Raton: CRC Press, 2012: FL47-66.
- [29] Denysiuk R, Costa L, Espírito Santo I. Clustering-based selection for evolutionary many-objective optimization[C]. Parallel Problem Solving from Nature — PPSN XIII. Cham: Springer, 2014: 538-547.
- [30] Deb K, Thiele L, Laumanns M, et al. Scalable test problems for evolutionary multiobjective[C]. Optimization Evolutionary Multiobjective Optimization. London: Springer, 2005: 105-145.
- [31] Cheng R, Li M, Tian Y. Benchmark functions for the CEC'2018 competition on many-objective optimization[C]. The IEEE Congress on Evolutionary Computation. Rio de Janeiro, 2018: 1-22.
- [32] 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.

作者简介

顾清华(1981—), 男, 教授, 博士生导师, 从事系统工程理论与应用、运输车辆调度及优化技术、复杂系统建模及仿真等研究, E-mail: qinghuagu@126.com;

张晓玥(1997—), 女, 硕士生, 从事智能优化算法的研究, E-mail: 724925536@qq.com;

陈露(1993—), 男, 博士生, 从事机器学习在智能矿山领域的研究, E-mail: chenlu@xauat.edu.cn.

(责任编辑: 李君玲)