

全局与局部模型交替辅助的差分进化算法

于成龙,付国霞,孙超利,张国晨

(太原科技大学 计算机科学与技术学院,太原 030024)

摘要: 为求解实际复杂工程应用中的高维计算费时优化问题,提出一种全局与局部代理模型交替辅助的差分进化算法。利用历史样本训练全局和局部代理模型,通过交替搜索全局和局部代理模型得到模型最优解并对其进行真实目标函数评价,实现探索和开采的平衡以减少真实目标函数的计算次数,同时通过针对性地选择个体进行真实目标函数计算,辅助算法快速找到目标函数的较优解。在15个低维测试问题和14个高维测试问题上的实验结果表明,在有限的计算资源情况下,该算法在12个低维测试问题上相较于最优重启策略代理辅助的社会学习粒子群优化算法、基于主动学习的代理模型辅助的粒子群优化算法等表现更好,在7个高维测试问题上相较于高斯过程辅助的进化算法、代理模型辅助的分层粒子群优化算法、求解高维费时问题的代理辅助的多种群优化算法等能找到目标函数的更优解。

关键词: 全局代理模型;局部代理模型;差分进化算法;计算费时优化问题;径向基函数网络

开放科学(资源服务)标志码(OSID):



中文引用格式: 于成龙,付国霞,孙超利,等.全局与局部模型交替辅助的差分进化算法[J].计算机工程,2022,48(3):115-123.

英文引用格式: YU C L, FU G X, SUN C L, et al. Differential evolution algorithm alternately assisted by global and local models[J]. Computer Engineering, 2022, 48(3): 115-123.

Differential Evolution Algorithm Alternately Assisted by Global and Local Models

YU Chenglong, FU Guoxia, SUN Chaoli, ZHANG Guochen

(School of Computer Science and Technology, Taiyuan University of Science and Technology, Taiyuan 030024, China)

[Abstract] To solve high-dimensional, computationally expensive optimization problems in practical complex engineering applications, a Differential Evolution (DE) algorithm assisted by the alternate optimization of global proxy and local proxy models is proposed. Therefore, a global or local proxy model is trained based on historical data and optimized alternately to assist the algorithm in finding the optimal solution of the original problem. A balance between exploration and exploitation can be achieved by alternately searching for the optimal solution of the global and local proxy models. The optimal solutions of global and local proxy models can be evaluated using the real objective function, which helps reduce the number of objective evaluations. Furthermore, it can accelerate the time required to find the optimal solution by selecting target individuals for real objective evaluation. The performance of the algorithm was tested on 15 low-dimensional and 14 high-dimensional test problems. Experimental results show that when computational resources are limited, the proposed algorithm performs better on 12 low-dimensional problems compared to the GORS-SSLPSO, CAL-SAPSO and other algorithms, and can find better optimal solutions for 7 high-dimensional problems than the MGP-SLPSO, SHPSO, SAMSO and other algorithms.

[Key words] global proxy model; local proxy model; Differential Evolution (DE) algorithm; computationally expensive optimization problem; Radial Basis Function Network (RBFN)

DOI: 10.19678/j.issn.1000-3428.0060693

0 概述

随着科学技术的快速发展,工农业生产、国防建

设、医疗卫生、人民生活等各个领域对工程产品性能的要求不断提高,工程优化问题也变得越来越复杂。元启发式优化算法(包括进化算法和群智能优化算

基金项目: 国家自然科学基金面上项目(61876123);山西省自然科学基金(201901D111264);山西省优秀人才科技创新项目(201805D211028)。

作者简介: 于成龙(1994—),男,硕士研究生,主研方向为计算智能;付国霞,硕士研究生;孙超利(通信作者),教授;张国晨,副教授。

收稿日期: 2021-01-25 **修回日期:** 2021-03-14 **E-mail:** chenglong_yu1994@qq.com

法)相比于传统的单点迭代算法对目标函数无连续可微的要求,具有更大的概率找到全局最优解,且不一定需要有显式的目标函数公式,在大型电力系统^[1-2]、天线设计^[3-4]、航空航天及汽车设计^[5-6]等科学研究和工程优化中具有广泛应用。然而,元启发式优化算法在获得最优解之前通常需要进行多次目标函数评价。因此,在目标函数计算费时的情况下,很难应用元启发式优化算法来搜索问题的最优设计方案^[7]。基于历史数据建立代理模型用来辅助元启发式优化算法,在评价次数有限的情况下搜索全局最优解是目前常用的求解计算费时问题的优化方法^[8-9]。相比于计算费时的原优化问题的目标函数评价,训练代理模型所需计算时间成本是低廉的,主要包括多项式回归(Polynomial Regression, PR)^[10]、径向基函数(Radial Basis Function, RBF)^[11]、高斯过程(Gaussian Process, GP)^[12]、人工神经网络(Artificial Neural Network, ANN)^[13-14]和支持向量回归(Support Vector Regression, SVR)^[15]等常见的代理模型。

元启发式优化算法中根据模型的用途可以将代理模型分为全局代理模型和局部代理模型。在通常情况下,全局代理模型用于平滑原问题的局部最优解,并引导搜索算法快速定位到最优解附近。LIU等^[16]提出一个求解中等规模费时优化问题的高斯过程辅助的进化算法。SCHNEIDER等^[17]采用高斯过程模型的贝叶斯优化以改进具有许多自由度的光学结构。TIAN等^[12]提出一个高斯过程辅助的进化算法(MGP-SLPSO),采用多目标填充准则的方式选择真实评价的个体。YU等^[18]提出一种最优重启策略代理辅助的社会学习粒子群优化算法(GORS-SLPSO),集成了重启策略和全局RBF代理模型以提供功能强大的优化器来解决计算耗时的问题。局部代理模型则用于较好地拟合原问题在某一个区域中的函数形状,从而辅助算法在该区域中进行搜索,从而提高最优解精度。LIM等^[7]使用集成和平滑的局部代理模型同时进行局部搜索,共同生成可靠的适应值预测和搜索改进策略。适应值继承是一种特殊的局部代理模型,SUN等^[19-20]基于同一代兄弟个体和父代个体提出一种新的适应值估计策略。全局和局部混合代理模型相比于使用单全局代理模型或者单局部代理模型在辅助优化算法搜索最优解时具有较好的搜索性能。WANG等^[21]提出基于主动学习的代理模型辅助的粒子群优化算法(CAL-SAPSO),结合全局和局部代理模型管理策略进行优化。SUN等^[11]提出代理模型辅助的协同优化算法,利用RBF全局代理模型辅助社会学习微粒群算法进行全局最优解的探索,同时利用适应值估计策略辅助的微粒群算法进行局部开采,从而实现对高维计算费时优化问题的求解。YU等^[22]提出代理模型辅助的分层粒子群优化算法(SHPSO),与代理模型辅助的协同优化算法不同,在SHPSO中,社会学习粒子群优化

用于进行局部空间的开采,而粒子群优化用于在全局空间中进行探索,从而提高全局最优解的寻找性能。LI等^[23]提出一种求解高维费时问题的代理辅助的多种群优化算法(SAMSO),全局和局部代理模型分别辅助两个种群进化,同时两个种群通过相互学习来提高算法的搜索性能。LIAO等^[24]利用所有历史样本训练全局代理模型,采用最优的若干样本训练局部代理模型,并通过多任务优化方法搜索两个模型的最优解。

混合代理模型辅助的元启发式优化算法能够利用不同模型的优势,相比于单代理模型能够更好地辅助优化算法在评价次数或者计算资源有限的情况下找到较优解。本文基于已有的历史数据,通过不同的样本选择策略建立全局代理模型和局部代理模型。对全局代理模型和局部代理模型进行最优解的交互搜索,通过对全局代理模型的搜索实现对原问题最优解的探索,模型最优解根据真实计算用于更新局部代理模型,使其能够更好地拟合目前最优解附近的局部区域,通过对局部代理模型最优解的搜索提高获得更优解的概率。

1 相关技术

1.1 径向基函数网络

径向基函数网络(Radial Basis Function Network, RBFN)是一种包含输入层、隐藏层和输出层共三层的神经网络。RBFN作为代理模型在辅助优化算法中具有较好的拟合效果,且其拟合效果对于问题维度的增加相对不敏感^[25-26]。给定样本库 $\{(x_i, f(x_i)), i=1, 2, \dots, N_a\}$, N_a 表示样本库中初始样本数量, RBFN模型可表示如下:

$$\hat{f}(x) = \sum_{i=1}^{N_c} \omega_i \times \varphi(\|x - c_i\|) \quad (1)$$

其中: $\omega = [\omega_1, \omega_2, \dots, \omega_{N_c}]$ 为RBF网络的输出权重; N_c 为隐含层节点数; $\varphi(\cdot)$ 为径向基核函数; c_i 为第 i 个隐含节点径向基函数的中心。常见的核函数有高斯核、线性核、多重二次曲面核、立方核等^[27]。本文算法中径向基核函数采用立方核函数。

1.2 差分进化算法

差分进化(Differential Evolution, DE)算法在许多优化问题上已被证明是一个简单有效的进化算法^[28-29],遵循进化算法的一般流程,即首先在问题决策空间中产生一个初始化种群 $\{x_1(t), x_2(t), \dots, x_N(t)\}$,其中, $t=0, N$ 为种群大小, $x_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ 表示第 i 个个体, D 表示问题的维度。然后通过变异、交叉和选择操作找到问题的最优解。在DE算法中,常见的变异策略有DE/rand/1、DE/current-to-best/1和DE/best/1等^[30]。本文算法中采用标准的DE/rand/1变异策略,即对于每一代 t ,随机选择3个不同于当前个体 i 的个体,并通过以下公式得到变异后的向量 $v_i(t)$:

$$\mathbf{v}_i(t+1) = \mathbf{x}_{r_0}(t) + F_i \cdot (\mathbf{x}_{r_1}(t) - \mathbf{x}_{r_2}(t)) \quad (2)$$

其中: r_0, r_1 和 r_2 为 $[1, N]$ 中3个不同的随机数; $\mathbf{x}_{r_0}, \mathbf{x}_{r_1}$ 和 \mathbf{x}_{r_2} 表示当前种群中不同于个体 \mathbf{x}_i 的3个个体; F_i 称为DE的变异因子。

每个个体根据一定的概率在各个维度上进行交叉操作,通过式(3)的交叉方法得到交叉后的向量 $\mathbf{u}_i(t+1) = (u_{i1}(t), u_{i2}(t), \dots, u_{id}(t))$ 。 $\mathbf{u}_i(t+1)$ 的计算公式如下:

$$u_{ij}(t+1) = \begin{cases} v_{ij}(t+1), & \text{rand} \leq CR_j, j = j_{\text{randint}} \\ x_{ij}(t), & \text{其他} \end{cases} \quad (3)$$

其中: $CR_j \in [0, 1]$ 为交叉概率; j_{randint} 为 $[1, D]$ 中随机产生的一个整数。对每个向量 $\mathbf{u}_i(t+1)$ 计算目标函数值,并与其父代个体进行适应值的对比,保留较优的个体作为下一代的父代个体,即:

$$\mathbf{x}_i(t+1) = \begin{cases} \mathbf{u}_i(t+1), & f(\mathbf{u}_i(t+1)) < f(\mathbf{x}_i(t)) \\ \mathbf{x}_i(t), & \text{其他} \end{cases} \quad (4)$$

2 全局与局部代理模型交替辅助的DE算法

2.1 算法整体流程

算法1给出了全局与局部代理模型交替辅助的差分进化算法AGL-DE的伪代码。首先,通过拉丁超立方体产生一定数量的初始样本,对其使用真实的目标函数评价并将其存入数据集Arc中。基于这些真实评价的个体确定当前最优解,记为gbest。然后,若没有达到停止条件,即达到最大的评价次数,则循环执行以下操作:交替建立全局代理模型和局部代理模型并对模型进行最优解搜索,搜索到的最优解进行真实目标函数的计算,存入数据集Arc中,并更新最优解gbest。最后,详细介绍全局代理模型和局部代理模型的训练(步骤6和步骤10)及其优化。

算法1 AGL-DE算法

输入 种群规模 N_p ,问题维度 D

输出 全局最优解

1. 初始化若干个样本点进行真实评价,并存入数据集Arc中;
2. 确定当前找到的最优位置,记为gbest;
3. model = 0;
4. 当不满足停止条件时,执行步骤5~步骤15,否则跳转到步骤16;
5. If model == 0
6. 训练全局代理模型;
7. 用DE搜索全局代理模型的最优解,并对找到的最优解 $\mathbf{x}_{\text{gbest}}$ 使用真实目标函数进行计算;
8. model = 1;
9. Else
10. 训练局部代理模型;
11. 用DE搜索局部代理模型的最优解,并对找到的最优解 $\mathbf{x}_{\text{ibest}}$ 使用真实目标函数进行计算;
12. model = 0;
13. EndIf
14. 将真实计算的个体存入数据集Arc中,并用该个体更新

gbest;

15. 返回步骤4;

16. 输出gbest及其目标函数值。

2.2 全局代理模型的训练与优化

在代理模型辅助的进化算法中,为了保证算法的收敛性,通常选用适应值最优的若干个样本来训练模型,但是这样会导致算法很快陷入局部最优,使得算法性能下降。本文选择具有代表性的样本建立全局代理模型,并通过在决策空间对全局代理模型最优解的搜索提高算法探索能力,防止算法陷入局部最优。文献[31]表明全局代理模型并非一定需要与原函数精确拟合,其主要作用在于平滑掉局部最优,加速定位到全局最优解附近。为此,本文通过计算每个样本 \mathbf{x} 和其最近邻个体之间的拥挤度(记为 $C(\mathbf{x}_i)$)来选择全局代理模型的训练样本。计算方式为对Arc中个体按照目标函数值从小到大排序,对任意个体 \mathbf{x}_i ,其拥挤度为相邻两个个体的目标函数之差。显然,拥挤度越大,说明该个体和其他个体之间的目标函数差值越大,拥挤度越小,说明和该个体的目标函数的相似度较大。为了更好地训练全局代理模型,选择拥挤度大且考虑估值小的样本个体来训练模型。通过-max的方式将最大化拥挤度转换成最小化问题,根据个体拥挤度和其适应值进行非支配排序,并从非支配排序后的第一层开始选择该层的所有样本放入训练样本集,直到当加入第 k 层的所有样本选择出的样本数大于预定样本数时,在第 k 层随机选择样本直到达到预定样本数,并利用选择出的所有样本训练全局代理模型。使用差分进化算法搜索模型的最优解,并对其进行真实的目标函数计算。需要注意的是对模型进行最优解搜索时,初始化种群中的个体来自:1)使用拉丁超立方体采样技术得到的初始个体;2)数据集Arc中适应值较优的若干个体。初始化种群中增加Arc中适应值较优的个体是为了加快模型最优解的搜索速度。将经过真实目标函数计算的模型最优解存入样本库Arc中并更新目前找到的最优解gbest。算法2为算法1中训练全局代理模型的伪代码。

算法2 全局代理模型训练算法

输入 数据集Arc,训练全局代理的样本数 N_g

输出 模型最优解 $\mathbf{x}_{\text{gbest}}$

1. 对Arc中所有个体进行适应值由小到大排序;
2. 计算每个个体 i 的拥挤度 C_i ;
3. 基于拥挤度和适应值对Arc中所有个体进行非支配排序;
4. 从第一个非支配面开始选取训练样本 N_s ,若已选样本数 N_s 和第 k 个非支配面上的样本数 N_k 之和超过所需训练样本数 N_t ,则从第 k 个非支配面上随机选取 $|N_t - N_s|$ 个样本;
5. 训练RBF代理模型 M_g ;
6. 初始化种群,记录种群中的最优解为 $\mathbf{x}_{\text{gbest}}$;
7. If 未达到停止条件 then
8. 对种群个体进行变异、交叉操作;
9. 使用 M_g 对每个个体进行估值;
10. 根据估值选择下一父代个体;

- 11.更新最优解 x_{gbest} ;
- 12.EndIf
- 13.输出最优解 x_{gbest} 。

2.3 局部代理模型的训练与优化

为快速找到更优解,利用数据集 Arc 中的若干最优解建立局部代理模型。算法 3 给出了训练局部代理模型的伪代码。与训练全局代理模型类似,从数据集 Arc 中选择一定数量的训练数据训练局部代理模型。利用差分进化算法进行局部代理模型最优解的搜索。对找到的局部代理模型最优解使用真实的目标函数进行计算,存入数据集 Arc 中并更新目前计算得到的最优解。

算法 3 局部代理模型训练算法

输入 数据集 Arc,训练局部代理模型的样本数 N_l

输出 模型最优解 x_{lbst}

- 1.对 Arc 中所有个体进行适应值由小到大排序;
- 2.按顺序选择 N_l 个训练样本;
- 3.训练 RBF 代理模型 M_l ;
- 4.初始化种群,记录种群中最优解为 x_{lbst} ;
- 5.If 未达到停止条件 then
- 6.对种群个体进行变异、交叉操作;
- 7.使用 M_l 对每个个体进行估值;
- 8.根据估值选择下一父代个体;
- 9.更新最优解 x_{lbst} ;
- 10.EndIf
- 11.输出最优解 x_{lbst} 。

3 实验设计与结果分析

为评估 AGL-DE 算法的性能,在 7 个单目标基准问题^[7,32]上进行测试,这 7 个基准问题的函数名称、全局最优值和函数特征如表 1 所示。本节首先给出了 AGL-DE 算法的参数设置,然后将其与近年来提出的求解费时问题的优化算法在测试函数上进行结果比较,最后对算法框架、参数进行敏感性分析。所有算法均独立运行 25 次,并通过显著性水平为 0.05 的带有 Bonferroni 校验的 Wilcoxon 秩和检验方法^[33]对 25 次独立运行结果进行显著性统计检验。

表 1 基准函数特性

Table 1 Characteristics of the benchmark functions

函数编号	函数名称	全局最优值	函数特征
F1	Ellipsoid Function	0	单模态
F2	Rosenbrock Function	0	多模态
F3	Ackley Function	0	多模态
F4	Griewank Function	0	多模态
F5	Rastrigin Function	0	多模态
F6	Shifted Rotated Rastrigin Function	-330	复杂多模态
F7	Rotated Hybrid Composition Function	10	复杂多模态

3.1 参数设置

在 AGL-DE 算法中,不同的代理模型均可作为全局与局部代理模型,本文采用 RBF 代理模型,主要原因在于 RBF 代理模型对于高维问题具有较好的拟合效果^[22]。训练 RBF 代理模型的最小样本数为 $2 \times D$ 。实验中初始产生的样本数为 $2 \times D$,随着评价次数增多,Arc 中真实计算的样本数增多。若样本过多,则训练模型的速度会减慢,因此在实验中设定训练模型的最大样本数为 $5 \times D$,在每次模型训练时训练模型的样本数为 $[2 \times D, 5 \times D]$ 中的一个随机整数 N_t 。需要注意的是若产生的随机整数 N_t 大于当前已有样本数,则 Arc 中所有样本均参与模型的训练,否则从 Arc 中随机挑选 N_t 个数据进行模型训练。对于模型最优解的搜索,初始种群大小设置为 50,从 Arc 中选择的最优解个数为 $[40, 50]$ 中的一个随机整数,剩余的初始解使用拉丁超立方体方法产生。迭代次数设置为 $[D, 4 \times D]$ 中产生的一个随机整数。使用 DE/rand/1 作为变异策略,二项式交叉作为 DE 的交叉策略,变异因子设置为 0.5,交叉概率设置为 0.3。在实验对比中,所有算法的终止条件均为达到最大的真实评价次数,在 10、20 和 30 维的低维测试问题上算法评价 $11 \times D$ 次,在 50 和 100 维的高维测试问题上算法评价 1 000 次。

3.2 与其他算法的实验结果对比分析

表 2 给出了 AGL-DE 与 GORS-SSLPSO^[18]、CAL-SAPSO^[21]、DDEA-SE^[34]、SHPSO^[22] 算法在低维测试问题上的统计结果,其中,符号“+”、“-”和“~”分别表示 AGL-DE 算法相较于其他算法具有显著性优、显著性差和没有显著性差异,加粗数据表示该数据对应的算法找到的解是所有算法中最优的,最后一行的数字分别表示 AGL-DE 算法相较于其他算法具有显著性优的个数、显著性差的个数和没有显著性差异的个数。SHPSO 算法在低维问题上只涉及了 30 维,因此仅对 SHPSO 算法在 30 维问题上的结果进行比较。从表 2 可以看出,除了 F5 测试问题以外,AGL-DE 算法在 F1~F4 这 4 个测试问题上基本优于其他算法,虽然 GORS-SSLPSO 算法在 F4 上的结果略好,但是与 AGL-DE 算法并没有显著性差异。

表 3 给出了 AGL-DE 与 GORS-SSLPSO^[18]、MGP-SLPSO^[12]、DDEA-SE^[34]、SHPSO^[22]、SAMSO^[23] 算法在高维测试问题上的统计结果。从表 3 中可以看出,AGL-DE 算法相比于 GORS-SSLPSO、MDP-SLPSO、DDEA-SE、SHPSO、SAMSO 算法在高维问题上获得了 13/14、8/14、13/14、10/14 和 9/14 的更好解。此外,还可以看出 AGL-DE 算法在 5/7 个 100 维测试问题上获得了比其他算法更优的结果,表明 AGL-DE 算法对于高维问题具有较好的求解效果。

表2 AGL-DE与GORS-SSLPSO、CAL-SAPSO、DDEA-SE、SHPSO算法在低维测试问题上的统计结果

Table 2 Statistical results of AGL-DE and GORS-SSLPSO, CAL-SAPSO, DDEA-SE, SHPSO algorithms on low-dimensional test problems

函数编号	维度	GORS-SSLPSO	CAL-SAPSO	DDEA-SE	SHPSO	AGL-DE
F1	10	4.93E-02(+)	3.60E-02(+)	9.38E-01(+)		2.35E-03
	20	2.32E-01(+)	2.75E+00(+)	2.40E+00(+)		2.53E-02
	30	3.91E-01(+)	1.98E+00(+)	4.39E+00(+)	9.03E+00(+)	1.04E-01
F2	10	1.64E+01(+)	1.60E+01(+)	2.56E+01(+)		8.92E+00
	20	4.87E+01(+)	3.71E+01(+)	4.39E+01(+)		1.90E+01
	30	9.05E+01(+)	5.51E+01(+)	5.64E+01(+)	4.88E+01(+)	2.91E+01
F3	10	3.83E+00(+)	1.88E+01(+)	5.84E+00(+)		2.56E+00
	20	4.00E+00(+)	1.89E+01(+)	5.07E+00(+)		2.88E+00
	30	5.72E+00(+)	1.54E+01(+)	4.72E+00(+)	5.17E+00(+)	2.97E+00
F4	10	7.51E-01(≈)	1.18E+00(+)	1.30E+00(+)		8.36E-01
	20	3.97E-01(+)	1.29E+00(+)	1.22E+00(+)		2.05E-01
	30	3.77E-01(+)	1.47E+00(+)	1.23E+00(+)	1.21E+00(+)	7.76E-02
F5	10	3.80E+01(-)	9.49E+01(≈)	5.23E+01(-)		8.69E+01
	20	4.78E+01(-)	8.05E+01(≈)	7.19E+01(≈)		1.27E+02
	30	8.06E+01(≈)	3.24E+01(-)	1.13E+02(+)	2.32E+02(+)	8.56E+01
+/-/≈		11/2/2	12/1/2	13/1/1	5/0/0	NAN

表3 AGL-DE与GORS-SSLPSO、MGP-SLPSO、DDEA-SE、SHPSO、SAMSO算法在高维测试问题上的统计结果

Table 3 Statistical results of AGL-DE and GORS-SSLPSO, MGP-SLPSO, DDEA-SE, SHPSO, SAMSO algorithms on high-dimensional test problems

函数编号	维度	GORS-SSLPSO	MGP-SLPSO	DDEA-SE	SHPSO	SAMSO	AGL-DE
F1	50	6.55E-01(+)	1.89E-06(-)	1.44E+01(+)	6.54E+00(+)	4.22E-01(+)	6.74E-03
	100	3.99E+01(+)	1.88E-02(-)	2.91E+02(+)	1.19E+02(+)	6.42E+01(+)	2.40E+00
F2	50	1.40E+02(+)	8.50E+01(+)	9.39E+01(+)	5.11E+01(+)	4.99E+01(+)	4.71E+01
	100	2.44E+02(+)	3.65E+02(+)	2.35E+02(+)	2.01E+02(+)	2.91E+02(+)	1.16E+02
F3	50	6.32E+00(+)	1.14E+01(+)	4.22E+00(+)	2.43E+00(-)	1.46E+00(-)	3.37E+00
	100	1.43E+01(+)	1.45E+01(+)	6.09E+00(+)	5.54E+00(≈)	6.07E+00(+)	5.51E+00
F4	50	1.52E-03(+)	9.91E-03(+)	1.95E+00(+)	9.64E-01(+)	6.75E-01(+)	5.12E-05
	100	1.39E-01(+)	6.90E-02(≈)	1.69E+01(+)	1.12E+00(+)	1.07E+00(+)	7.64E-02
F5	50	1.55E+02(+)	8.79E+01(-)	1.74E+02(+)	3.36E+02(+)	3.95E+01(-)	1.22E+02
	100	3.22E+02(+)	7.30E+02(+)	8.27E+02(+)	8.74E+02(+)	4.69E+02(+)	2.13E+02
F6	50	-3.76E+01(+)	-1.43E+02(-)	7.10E+02(+)	1.21E+02(+)	-1.60E+02(-)	-8.18E+01
	100	4.02E+02(+)	5.22E+02(+)	1.81E+03(+)	6.96E+02(+)	5.44E+02(+)	2.23E+02
F7	50	1.07E+03(≈)	1.04E+03(≈)	1.19E+03(+)	1.00E+03(-)	9.68E+02(-)	1.05E+03
	100	1.29E+03(+)	1.30E+03(+)	1.25E+03(≈)	1.18E+03(≈)	1.14E+03(-)	1.21E+03
+/-/≈		13/0/1	8/4/2	13/0/1	10/2/2	9/5/0	NAN

为更好地分析算法的收敛性能,图1、图2分别给出了AGL-DE算法和其他算法在低维和高维测试问题上的函数收敛曲线图,横坐标代表真实适应值的评估次数,记为 T ,为了能够更加明显地观察收敛曲线的变化情况,纵坐标用25次独立运行得到的

适应值的中值的对数表示,记为 $\lg(F(\mathbf{x}_T))$,其中 \mathbf{x}_T 为算法第 T 次真实适应值评估下找到的最优解, $F(\mathbf{x}_T)$ 为 \mathbf{x}_T 对应的目标函数值。

由于F6测试问题的适应值可能是负值,因此在该测试问题的收敛曲线图上没有对适应值进行取对

数操作。需要注意的是,DDEA-SE算法是一种基于数据驱动的离线优化算法,在优化的过程中不会产

生需要真实适应度评估的个体,故没有给出其收敛曲线图。

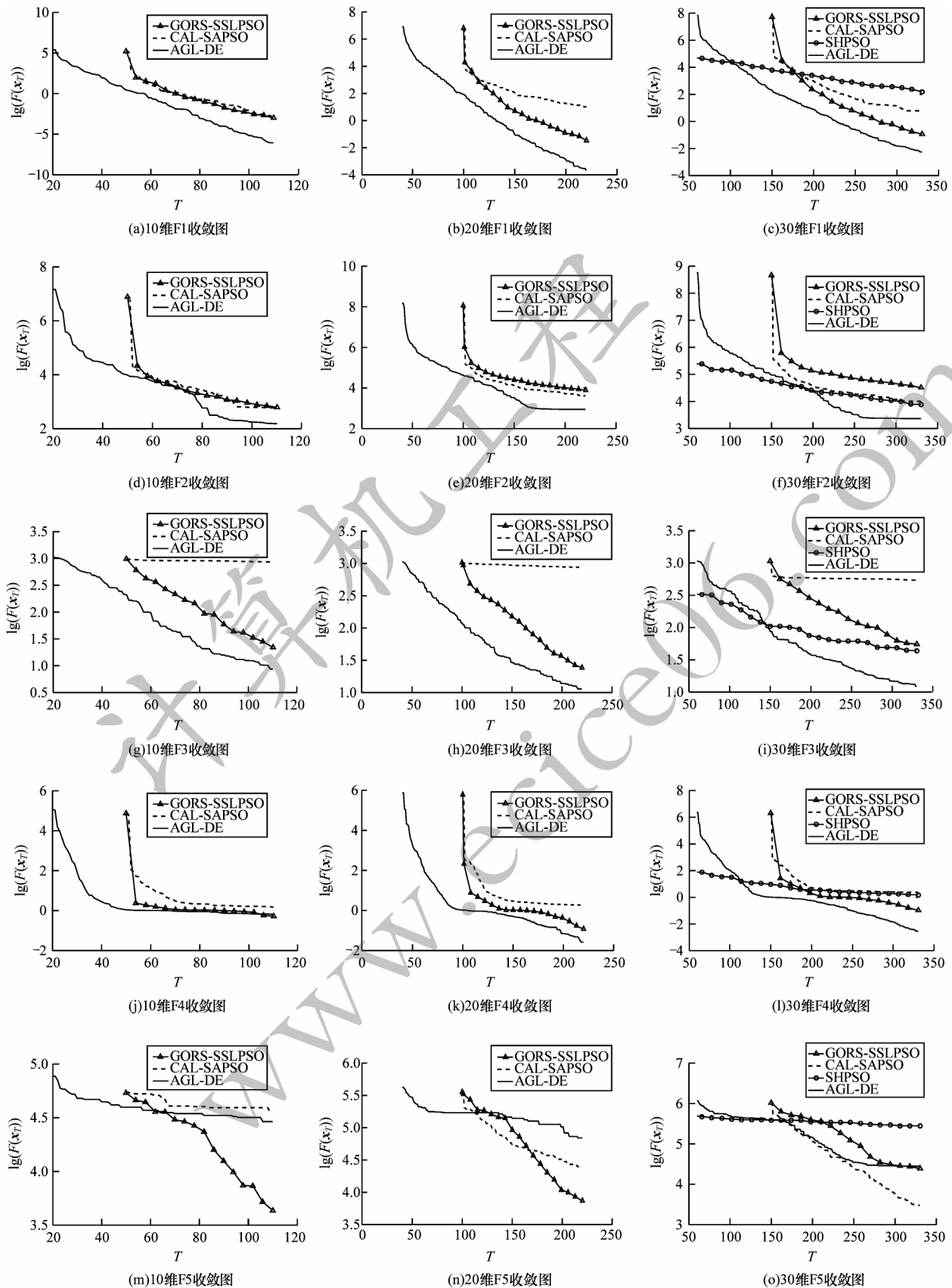


图1 AGL-DE与GORS-SSLPSO、CAL-SAPSO、SHPSO算法在低维测试问题上的函数收敛曲线图

Fig.1 Function convergence curves of AGL-DE and GORS-SSLPSO, CAL-SAPSO, SHPSO algorithms on low-dimensional test problems

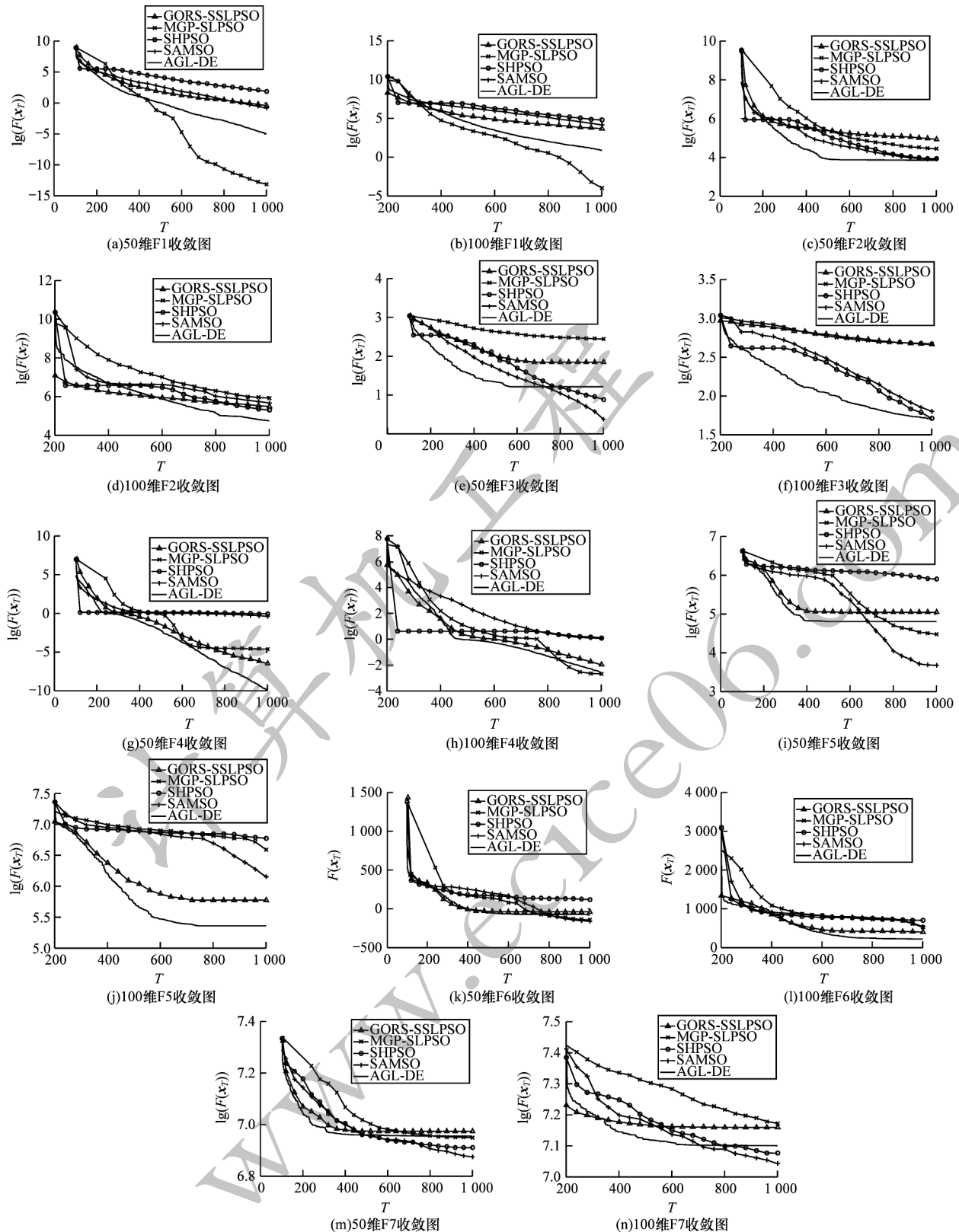


图2 AGL-DE与GORS-SSLPSO、MGP-SLPSO、SHPSO、SAMSO算法在高维测试问题上的函数收敛曲线图

Fig.2 Function convergence curves of AGL-DE and GORS-SSLPSO, MGP-SLPSO, SHPSO, SAMSO algorithms on high-dimensional test problems

从图1可以看出,AGL-DE在F1~F4测试问题上都比其他算法具有更快的收敛速度,在低维的单模态和多模态问题上都展现了良好的性能。对于高维优化问题,从图2可以看出,本文所提AGL-DE算法

在大部分优化问题上具有较快的收敛速度。但是从图2也可以看出其容易陷入局部最优,如50维的F2、F3、F5和F7测试问题,在搜索算法到达一定程度后不再继续下降。由此分析其早熟收敛的主要原因在

于初始样本的选择。从Arc中选择若干最优解作为DE的初始种群,当这些解的差别较小时,初始种群的多样性较差。这些解虽然能够引导算法快速收敛到某一个解,但是很难跳出局部最优,从而导致了算法的早熟收敛。

3.3 全局与局部代理模型有效性分析

为验证全局与局部代理模型交替辅助的有效性,将其与仅使用全局代理模型辅助进化算法(G-DE)和仅使用局部代理模型辅助进化算法(L-DE)进行实验结果比较,在实验过程中其他参数设置与3.1节的设置相同。表4给出了AGL-DE、G-DE和L-DE算法在10维、20维和50维的F1、F3和F4测试问题上的统计结果。从统计结果上来看:AGL-DE比G-DE显著性好3个,显著性差1个,无显著性差异5个;AGL-DE比L-DE显著性好3个,显著性差2个,无显著性差异4个。统计结果表明:全局代理模型和局部代理模型交替辅助差分进化算法比单一模型辅助差分进化算法的效果好。

表4 AGL-DE与G-DE、L-DE算法在F1、F3和F4测试问题上的统计结果

Table 4 Statistical results of AGL-DE and G-DE, L-DE algorithms on F1, F3 and F4 test problems

函数编号	维度	G-DE	L-DE	AGL-DE
F1	10	1.94E-02(+)	7.63E-03(+)	2.35E-03
	30	1.97E-01(≈)	6.40E-02(≈)	1.04E-01
	50	3.57E-02(+)	3.98E-03(-)	6.74E-03
F3	10	3.43E+00(+)	1.31E+00(-)	2.56E+00
	30	2.96E+00(≈)	2.98E+00(≈)	2.97E+00
	50	2.68E+00(-)	4.00E+00(+)	3.37E+00
F4	10	8.79E-01(≈)	9.06E-01(≈)	8.36E-01
	30	6.56E-02(≈)	1.29E-01(+)	7.76E-02
	50	6.66E-05(≈)	4.19E-05(≈)	5.12E-05
+/-/≈		3/1/5	3/2/4	NAN

3.4 参数敏感性分析

为验证在一定区间内随机产生的训练集样本数 N_t 、种群迭代次数iter和从Arc中选择的初始样本数 N_a 的有效性,在不改变其他参数设置的情况下,与固定这3个参数的3种策略进行比较,分别将它们固定为区间内的最小值、中值和最大值,即min-values、med-values和max-values,具体参数设置如表5所示。在这3种策略上对50维和100维的F1、F2、F4和F6测试问题进行测试,表6给出了AGL-DE算法和这3种策略的对比统计结果,可以看出在一定区间内随机生成参数的AGL-DE算法要比单纯固定这些参数的3种策略具有更好的性能。

表5 3种策略的参数设置

Table 5 Parameter settings of three strategies

策略名称	N_t	iter	N_a
min-values	$2 \times D$	D	40
med-values	$3.5 \times D$	$2.5 \times D$	45
max-values	$5 \times D$	$4 \times D$	50

表6 AGL-DE算法与min-values、med-values、max-values策略在F1、F2、F4和F6测试问题上的统计结果

Table 6 Statistical results of AGL-DE algorithm and min-values, med-values, max-values strategies on F1, F2, F4 and F6 test problems

函数编号	维度	min-values	med-values	max-values	AGL-DE
F1	50	1.60E-01(+)	2.22E-02(+)	7.47E-03(≈)	6.74E-03
	100	1.82E+01(+)	5.08E+00(+)	4.56E+00(+)	2.40E+00
F2	50	9.02E+01(+)	4.74E+01(≈)	4.90E+01(+)	4.71E+01
	100	2.13E+02(+)	2.14E+02(+)	1.81E+02(+)	1.16E+02
F4	50	8.58E-05(+)	6.19E-05(≈)	1.09E-04(+)	5.12E-05
	100	9.99E-02(+)	1.30E-01(+)	1.38E-01(+)	7.64E-02
F6	50	2.47E+02(+)	-6.14E+01(≈)	-4.96E+01(≈)	-8.18E+01
	100	4.44E+02(≈)	2.19E+02(≈)	2.54E+02(≈)	2.23E+02
+/-/≈		7/0/1	4/0/4	5/0/3	NAN

4 结束语

本文提出一种全局与局部代理模型交替辅助的差分进化算法,通过对全局代理模型和局部代理模型最优解的交替搜索来更新数据集以及原费时优化问题的最优解,从而在计算资源有限的情况下获得问题的更优解。实验结果表明,相比于近年来提出的求解费时问题的优化算法,该算法在低维和高维测试问题上均获得了较好的寻优效果,然而在某些高维测试问题上容易陷入局部最优,导致早熟收敛。因此在下一步工作中将考虑增强算法多样性,以便跳出局部最优,从而在计算资源有限的情况下找到目标函数的更优解。

参考文献

- [1] DEL VALLE Y, VENAYAGAMOORTHY G K. Particle swarm optimization: basic concepts, variants and applications in power systems[J]. IEEE Transactions on Evolutionary Computation, 2008, 12(2): 171-195.
- [2] 吕铭晟,沈洪远,李志高,等. 多变异策略差分进化算法的研究与应用[J]. 计算机工程, 2014, 40(12): 146-150. LÜ M S, SHEN H Y, LI Z G, et al. Research and application of differential evolution algorithm under multiple mutation strategy [J]. Computer Engineering, 2014, 40(12): 146-150. (in Chinese)
- [3] SOLTANI S, MURCH R D. A compact planar printed MIMO antenna design[J]. IEEE Transactions on Antennas and Propagation, 2015, 63(3): 1140-1149.
- [4] MILLIGAN T A. Modern antenna design[M]. Hoboken, USA: John Wiley & Sons, Inc., 2005.
- [5] REGIS R G. Evolutionary programming for high-dimensional constrained expensive black-box optimization using radial basis functions[J]. IEEE Transactions on Evolutionary Computation, 2014, 18(3): 326-347.
- [6] ONG Y S, NAIR P B, KEANE A J. Evolutionary optimization of computationally expensive problems via surrogate modeling [J]. AIAA Journal, 2003, 41(4): 687-696.
- [7] LIM D, JIN Y C, ONG Y S, et al. Generalizing surrogate-

- assisted evolutionary computation[J]. IEEE Transactions on Evolutionary Computation, 2010, 14(3): 329-355.
- [8] QUEIPO N V, HAFKAR T, SHYY W, et al. Surrogate-based analysis and optimization[J]. Progress in Aerospace Sciences, 2005, 41(1): 1-28.
- [9] WORTMANN T, COSTA A, NANNICINI G, et al. Advantages of surrogate models for architectural design optimization[J]. Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 2015, 29(4): 471-481.
- [10] TANG Y F, CHEN J Q, WEI J H. A surrogate-based particle swarm optimization algorithm for solving optimization problems with expensive black box functions[J]. Engineering Optimization, 2013, 45(5): 557-576.
- [11] SUN C L, JIN Y C, CHENG R, et al. Surrogate-assisted cooperative swarm optimization of high-dimensional expensive problems[J]. IEEE Transactions on Evolutionary Computation, 2017, 21(4): 644-660.
- [12] TIAN J, TAN Y, ZENG J C, et al. Multiobjective infill criterion driven Gaussian process-assisted particle swarm optimization of high-dimensional expensive problems[J]. IEEE Transactions on Evolutionary Computation, 2019, 23(3): 459-472.
- [13] 张宗华,赵京湘,卢享,等. 基于遗传算法的BP神经网络在电力负载预测中的应用[J]. 计算机工程, 2017, 43(10): 277-282, 288.
- ZHANG Z H, ZHAO J X, LU X, et al. Application of BP neural network based on genetic algorithm in power load forecasting[J]. Computer Engineering, 2017, 43(10): 277-282, 288. (in Chinese)
- [14] JIN Y C, OLHOFFER M, SENDHOFF B. A framework for evolutionary optimization with approximate fitness functions[J]. IEEE Transactions on Evolutionary Computation, 2002, 6(5): 481-494.
- [15] JIN Y. A comprehensive survey of fitness approximation in evolutionary computation[J]. Soft Computing, 2005, 9(1): 3-12.
- [16] LIU B, ZHANG Q F, GIELEN G G E. A Gaussian process surrogate model assisted evolutionary algorithm for medium scale expensive optimization problems[J]. IEEE Transactions on Evolutionary Computation, 2013, 18(2): 180-192.
- [17] SCHNEIDER P I, SANTIAGO X G, ROCKSTUHL C, et al. Global optimization of complex optical structures using Bayesian optimization based on Gaussian processes[EB/OL]. [2020-12-15]. <http://arxiv.org/pdf/1707.08479>.
- [18] YU H B, TAN Y, SUN C L, et al. A generation-based optimal restart strategy for surrogate-assisted social learning particle swarm optimization[J]. Knowledge-Based Systems, 2019, 163: 14-25.
- [19] SUN C L, ZENG J C, PAN J, et al. A new fitness estimation strategy for particle swarm optimization[J]. Information Sciences, 2013, 221: 355-370.
- [20] SUN C L, DING J L, ZENG J C, et al. A fitness approximation assisted competitive swarm optimizer for large scale expensive optimization problems[J]. Memetic Computing, 2018, 10(2): 123-134.
- [21] WANG H D, JIN Y C, DOHERTY J. Committee-based active learning for surrogate-assisted particle swarm optimization of expensive problems[J]. IEEE Transactions on Cybernetics, 2017, 47(9): 2664-2677.
- [22] YU H B, TAN Y, ZENG J C, et al. Surrogate-assisted hierarchical particle swarm optimization[J]. Information Sciences, 2018, 454/455: 59-72.
- [23] LI F, CAI X, GAO L, et al. A surrogate-assisted multiswarm optimization algorithm for high-dimensional computationally expensive problems[J]. IEEE Transactions on Cybernetics, 2021, 51(3): 1390-1402.
- [24] LIAO P, SUN C L, ZHANG G C, et al. Multi-surrogate multi-tasking optimization of expensive problems[EB/OL]. [2020-12-15]. https://www.researchgate.net/publication/343051439_Multi-surrogate_multi-tasking_optimization_of_expensive_problems.
- [25] 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.
- [26] KATTAN A, GALVAN E. Evolving radial basis function networks via GP for estimating fitness values using surrogate models[C]//Proceedings of 2012 IEEE Congress on Evolutionary Computation. Washington D. C., USA: IEEE Press, 2012: 1-7.
- [27] GUTMANN H M. A radial basis function method for global optimization[J]. Journal of Global Optimization, 2001, 19(3): 201-227.
- [28] STORN R, PRICE K. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces[J]. Journal of Global Optimization, 1997, 11(4): 341-359.
- [29] 代瑞瑞,马永杰,摆玉龙,等. 基于动态模式搜索的差分进化算法[J]. 计算机工程, 2016, 42(9): 163-167.
- DAI R R, MA Y J, BAI Y L, et al. Differential evolution algorithm based on dynamic pattern search[J]. Computer Engineering, 2016, 42(9): 163-167. (in Chinese)
- [30] ZHANG J Q, SANDERSON A C. JADE: adaptive differential evolution with optional external archive[J]. IEEE Transactions on Evolutionary Computation, 2009, 13(5): 945-958.
- [31] SUN C L, JIN Y C, ZENG J C, et al. A two-layer surrogate-assisted particle swarm optimization algorithm[J]. Soft Computing, 2015, 19(6): 1461-1475.
- [32] SUGANTHAN P N, HANSEN N, LIANG J J, et al. Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization[EB/OL]. [2020-12-15]. http://www.al-roomi.org/multimedia/CEC_Database/CEC2013/RealParameterOptimization/CEC2013_RealParameterOptimization_TechnicalReport.pdf.
- [33] HENLEY S. Principles and procedure of statistics: a biometrical approach[J]. Computers & Geosciences, 1983, 9(2): 275.
- [34] WANG H D, JIN Y C, SUN C L, et al. Offline data-driven evolutionary optimization using selective surrogate ensembles[J]. IEEE Transactions on Evolutionary Computation, 2019, 23(2): 203-216.