PlatEMO

进化多目标优化平台

用户手册 4.0

生物智能与知识发现（BIMK）研究所

2022 年 10 月 15 日

非常感谢使用由安徽大学生物智能与知识发现（BIMK）研究所开发的进化多目标优化平台 PlatEMO。本平台是一个开源免费的代码库，仅供教学与科研使用，不得用于商业用途。本平台中的代码基于作者对论文的理解编写而成，作者不对用户因使用代码产生的任何后果负责。包含利用本平台产生的数据的论文应在正文中声明对 PlatEMO 的使用，并引用以下参考文献：

Ye Tian, Ran Cheng, Xingyi Zhang, and Yaochu Jin, “PlatEMO: A MATLAB platform for evolutionary multi-objective optimization [educational forum],” IEEE Computational Intelligence Magazine, 2017, 12(4): 73-87.

如有任何意见或建议，欢迎联系 field910921@gmail.com（田野）。如想将您的代码添加进 PlatEMO 中并公开，也欢迎联系 field910921@gmail.com。您可以在 GitHub 上获取 PlatEMO 的最新版本。

目　录

一　快速入门..1

二　通过命令行使用 PlatEMO..3

1.　求解测试问题..3

2.　求解自定义问题..5

3.　获取运行结果..8

三　通过图形界面使用 PlatEMO..10

1.　测试模块..10

2.　应用模块..11

3.　实验模块..12

4.　算法、问题和指标的标签..12

四　扩展 PlatEMO..15

1.　算法类..15

2.　问题类..17

3.　个体类..23

4.　一次完整的运行过程..24

5.　指标函数..25

五　算法列表..27

六　问题列表..35

一 快速入门

软件要求：MATLAB R2018a 或以上（不使用 PlatEMO 图形界面）或
MATLAB R2020b 或以上（使用 PlatEMO 图形界面）及
并行计算工具箱 和
统计与机器学习工具箱

PlatEMO 是一个用于求解优化问题的开源平台，它的输入是一个优化问题，输出是在该优化问题上得到的最优解。一个优化问题满足以下定义：

$$\begin{aligned}\min_{\mathbf{x}} \quad & \mathbf{f}(\mathbf{x}) = \left(f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_M(\mathbf{x})\right) \\ \mathrm{s.t.} \quad & \mathbf{x} = (x_1, x_2, \ldots x_D) \in \Omega \\ & g_1(\mathbf{x}), g_2(\mathbf{x}), \ldots, g_K(\mathbf{x}) \leq 0\end{aligned}$$

其中 $\mathbf{x}$ 表示该问题的一个解或决策向量，它由 D 个决策变量 x_i 组成，其中每个决策变量可能被限制为实数、整数或二进制数等。Ω 表示该问题的搜索空间，它由下界 $l_1, l_2, \ldots l_D$ 和上界 $u_1, u_2, \ldots u_D$ 构成，即任意决策变量始终满足 $l_i \leq x_i \leq u_i$。$f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_M(\mathbf{x})$ 表示该解的 M 个目标函数值，$g_1(\mathbf{x}), g_2(\mathbf{x}), \ldots, g_K(\mathbf{x})$ 表示该解的 K 个约束违反值。

为了定义一个优化问题，用户至少需要输入以下内容：

- 决策变量的数目 D 和目标函数的数目 M；
- 每个决策变量的编码方式（实数、整数或二进制数等）；
- 决策变量的下界 $l_1, l_2, \ldots l_D$ 和上界 $u_1, u_2, \ldots u_D$；
- 至少一个目标函数 $f_1(\mathbf{x})$。

为了更精准地定义问题，用户还能输入以下内容：

- 多个目标函数 $f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_M(\mathbf{x})$；
- 多个约束函数 $g_1(\mathbf{x}), g_2(\mathbf{x}), \ldots, g_K(\mathbf{x})$；
- 解的初始化函数；
- 无效解的修复函数；
- 解的评价函数；

- 目标函数的梯度函数 $f_1'(\mathbf{x}), f_2'(\mathbf{x}), \dots, f_M'(\mathbf{x})$；
- 约束函数的梯度函数 $g_1'(\mathbf{x}), g_2'(\mathbf{x}), \dots, g_K'(\mathbf{x})$；
- 各函数计算中使用到的数据（一个任意类型的常量）。

以上函数均指的是代码函数而非数学函数，即它需要有符合规定的输入和输出，但不需要有显式的数学表达式。此外，用户还能定义与优化算法相关的内容，通过选择合适的算法和参数设置以提升优化效果。

在 MATLAB 中，用户可以用以下三种方式运行主函数文件 `platemo.m`：

1) 带参数调用主函数：

```
platemo('problem',@SOP_F1,'algorithm',@GA,'Name',Value,…);
```

可以利用指定的算法来求解指定的测试问题并设置参数，求解结果可以被显示在窗口中、保存在文件中或作为函数返回值（参阅求解测试问题章节）。

2) 带参数调用主函数：

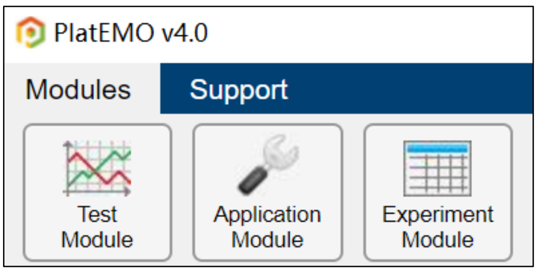
```
f1 = @(x)sum(x);
g1 = @(x)1-sum(x);
platemo('objFcn',f1,'conFcn',g1,'algorithm',@GA,…);
```

可以利用指定的算法来求解自定义的问题（参阅求解自定义问题章节）。

3) 不带参数调用主函数：

```
platemo();
```

可以弹出一个带有三个模块的图形界面，其中测试模块用于可视化地研究单个算法在单个问题上的性能（参阅测试模块章节），应用模块用于求解自定义问题（参阅应用模块章节），实验模块用于统计分析多个算法在多个问题上的性能（参阅实验模块章节）。



二 通过命令行使用 PlatEMO

1. 求解测试问题

用户可以以如下形式带参数调用主函数 platemo()来求解测试问题：

```
platemo('Name1',Value1,'Name2',Value2,'Name3',Value3,…);
```

其中所有可接受的参数列举如下：

参数名	数据类型	默认值	描述
'algorithm'	函数句柄或单元数组	不定	要运行的算法类
'problem'	函数句柄或单元数组	不定	要求解的问题类
'N'	正整数	100	种群大小
'M'	正整数	不定	问题的目标数
'D'	正整数	不定	问题的变量数
'maxFE'	正整数	10000	最大评价次数
'maxRuntime'	正数	inf	最大运行时间
'save'	整数	-10	保存的种群数
'outputFcn'	函数句柄	@DefaultOutput	每代开始前调用的函数 输入一：ALGORITHM 对象 输入二：PROBLEM 对象 输出：无

- 'algorithm'表示待运行的算法，它的值可以是一个算法类的句柄，例如@GA。它的值还可以是形如{@GA,p1,p2,…}的单元数组，其中 p1,p2,… 指定了该算法中的参数值。例如以下代码用算法@GA 求解默认问题，并设置了该算法中的参数值：

```
platemo('algorithm',{@GA,1,30,1,30});
```

- 'problem'表示待求解的测试问题，它的值可以是一个问题类的句柄，例如@SOP_F1。它的值还可以是形如{@SOP_F1,p1,p2,…}的单元数组，其中 p1,p2,… 指定了该问题中的参数值。例如以下代码用默认算法求解问题@WFG1，并设置了该问题中的参数值：

```
platemo('problem',{@WFG1,20});
```

- 'N'表示算法使用的种群的大小，它通常等于最终输出的解的个数。例如以下代码用算法@GA 求解问题@SOP_F1，并设置种群大小为 50：

```
platemo('algorithm',@GA,'problem',@SOP_F1,'N',50);
```

- 'M'表示问题的目标个数，它仅对一些多目标测试问题生效。例如以下代码用算法@NSGAII 求解具有 5 个目标的@DTLZ2 问题：

```
platemo('algorithm',@NSGAII,'problem',@DTLZ2,'M',5);
```

- 'D'表示问题的变量个数，它仅对一些测试问题生效。例如以下代码用算法@GA 求解具有 100 个变量的@SOP_F1 问题：

```
platemo('algorithm',@GA,'problem',@SOP_F1,'D',100);
```

- 'maxFE'表示算法可用的最大评价次数，它通常等于种群大小乘以迭代次数。例如以下代码设置算法@GA 的最大评价次数为 20000：

```
platemo('algorithm',@GA,'problem',@SOP_F1,'maxFE',20000);
```

- 'maxRuntime'表示算法可用的最大运行时间，单位为秒。当 maxRuntime 等于默认值 inf 时，算法将在 maxFE 次评价次数后停止；否则，算法将在 maxRuntime 秒后停止。例如以下代码设置算法@GA 的最大运行时间为 10 秒：

```
platemo('algorithm',@GA,'problem',@SOP_F1,'maxRuntime',10);
```

- 'save'表示保存的种群数，该值大于零时优化结果将被保存在文件中，该值小于零时优化结果将被显示在窗口中（参阅获取运行结果章节）。
- 'outputFcn'表示算法每代开始前调用的函数。该函数必须有两个输入和零个输出，其中第一个输入是当前的 ALGORITHM 对象、第二个输入是当前的 PROBLEM 对象。默认的'outputFcn'会根据当前'save'的值来保存或显示优化结果。

注意以上每个参数均有一个默认值，用户可以在调用时省略任意参数。

2. 求解自定义问题

当不指定参数'problem'时，用户可以通过指定以下参数来自定义问题：

参数名	数据类型	默认值	描述
'objFcn'	函数句柄或单元数组	{}	问题的目标函数；所有目标函数均被最小化 输入：一个决策向量 输出：目标值（标量）
'encoding'	标量或行向量	1	每个变量的编码方式
'lower'	标量或行向量	0	每个变量的下界
'upper'	标量或行向量	1	每个变量的上界
'conFcn'	函数句柄或单元数组	{}	问题的约束函数；当且仅当约束违反值小于等于零时，该约束被满足 输入：一个决策向量 输出：约束违反值（标量）
'decFcn'	函数句柄	{}	无效解修复函数 输入：一个决策向量 输出：修复后的决策向量
'evalFcn'	函数句柄	{}	解的评价函数 输入：一个决策向量 输出一：修复后的决策向量 输出二：所有目标值（向量） 输出三：所有约束违反值（向量）
'initFcn'	函数句柄	{}	种群初始化函数 输入：种群大小 输出：种群的决策向量构成的矩阵
'objGradFcn'	函数句柄或单元数组	{}	目标函数的梯度函数 输入：一个决策向量 输出：梯度（向量）
'conGradFcn'	函数句柄或单元数组	{}	约束函数的梯度函数 输入：一个决策向量 输出：梯度（向量）
'data'	任意	{}	问题的数据

- 'objFcn'表示问题的目标函数，它的值可以是一个函数句柄（单目标）或一个单元数组（多目标）。每个目标函数必须有一个输入和一个输出，其中输入是一个决策向量、输出是目标值。所有目标函数均被最小化。例如以下代码利用默认算法求解一个双目标优化问题：

```
f1 = @(x)x(1)+sum(x(2:end));
f2 = @(x)sqrt(1-x(1)^2)+sum(x(2:end));
platemo('objFcn',{f1,f2});
```

其中第一个目标为 $x_1+\sum_{i=2}^{D} x_i$、第二个目标为 $\sqrt{1-x_1^2}+\sum_{i=2}^{D} x_i$。

- 'encoding'表示每个变量的编码方式，它的值是一个标量或行向量，且每维的值可以为 1（实数）、2（整数）、3（标签）、4（二进制数）或 5（序列编号）。算法针对不同的编码方式可能使用不同的算子来产生解。例如以下代码指定三个实数变量、两个整数变量以及一个二进制变量：

```
f1 = @(x)x(1)+sum(x(2:end));
f2 = @(x)sqrt(1-x(1)^2)+sum(x(2:end));
platemo('objFcn',{f1,f2},'encoding',[1,1,1,2,2,4]);
```

问题的变量数 D 将根据'encoding'的长度自动确定。

- 'lower'和'upper'分别表示每个变量的下界和上界，它们的值是标量或行向量，且每维的值必须为实数。'lower'和'upper'的长度必须与'encoding'相同。例如以下代码指定搜索空间为 $[0,1]\times[0,9]^5$：

```
f1 = @(x)x(1)+sum(x(2:end));
f2 = @(x)sqrt(1-x(1)^2)+sum(x(2:end));
platemo('objFcn',{f1,f2},'encoding',[1,1,1,2,2,4],...
'lower',0,'upper',[1,9,9,9,9,9]);
```

- 'conFcn'表示问题的约束函数，它的值可以是一个函数句柄（单约束）或一个单元数组（多约束）。每个约束函数必须有一个输入和一个输出，其中输入是一个决策向量、输出是约束违反值。当且仅当约束违反值小于等于零时，该约束被满足。例如以下代码利用默认算法求解一个双目标优化问题：

```
f1 = @(x)x(1)+sum(x(2:end));
f2 = @(x)sqrt(1-x(1)^2)+sum(x(2:end));
g1 = @(x)1-sum(x(2:end));
platemo('objFcn',{f1,f2},'encoding',[1,1,1,2,2,4],...
'conFcn',g1,'lower',0,'upper',[1,9,9,9,9,9]);
```

并添加约束函数 $\sum_{i=2}^{6} x_i \geq 1$。注意，等式约束必须转换为不等式约束来处理。

- 'decFcn'表示问题的无效解修复函数，它的值必须是一个函数句柄。该函数必须有一个输入和一个输出，其中输入是一个决策向量、输出是修复后的决策向量。例如以下代码限制 x_1 为 0.1 的倍数：

```
f1 = @(x)x(1)+sum(x(2:end));
```

```
f2 = @(x)sqrt(1-x(1)^2)+sum(x(2:end));
g1 = @(x)1-sum(x(2:end));
h = @(x)[round(x(1)/0.1)*0.1,x(2:end)];
platemo('objFcn',{f1,f2},'encoding',[1,1,1,2,2,4],...
'conFcn',g1,'decFcn',h,'lower',0,'upper',[1,9,9,9,9,9]);
```

- 'evalFcn'表示解的评价函数，它的值必须是一个函数句柄。该函数必须有一个输入和三个输出，其中输入是一个决策向量、第一个输出是修复后的决策向量、第二个输出是目标值向量、第三个输出是约束违反值向量。默认的'evalFcn'通过依次调用'decFcn'、'objFcn'和'conFcn'来评价解，而以下代码定义了一个新的'evalFcn'来同时进行解的修复、目标计算和约束计算：

```
function [x,f,g] = Eval(x)
    x = [round(x(1)/0.1)*0.1,x(2:end)];
    x = max(0,min([1,9,9,9,9,9],x));
    f(1) = x(1)+sum(x(2:end));
    f(2) = sqrt(1-x(1)^2)+sum(x(2:end));
    g = 1-sum(x(2:end));
end
```

接着，以下代码通过仅指定评价函数定义了相同的问题：

```
platemo('evalFcn',@Eval,'encoding',[1,1,1,2,2,4],...
'lower',0,'upper',[1,9,9,9,9,9]);
```

- 'initFcn'表示种群初始化函数，它的值必须是一个函数句柄。该函数必须有一个输入和一个输出，其中输入是种群大小、输出是种群的决策向量构成的矩阵。默认的'initFcn'在整个搜索空间内随机产生初始解，而以下代码定义了一个新的'initFcn'以加速收敛：

```
q = @(N)rand(N,6);
platemo('evalFcn',@Eval,'encoding',[1,1,1,2,2,4],...
'initFcn',q,'lower',0,'upper',[1,9,9,9,9,9]);
```

- 'objGradFcn'和'conGradFcn'分别表示目标函数和约束函数的梯度函数，它们的值可以是一个函数句柄或一个单元数组。每个梯度函数必须有一个输入和一个输出，其中输入是一个决策向量、输出是梯度。默认的梯度函数通过有限差分来估计梯度，而以下代码定义了一个新的'objGradFcn'以加速收敛并保证种群的多样性：

```
fg = @(x)[0,x(2:end)];
platemo('evalFcn',@Eval,'encoding',[1,1,1,2,2,4],...
'objGradFcn',fg,'lower',0,'upper',[1,9,9,9,9,9]);
```

注意仅有少量算法会使用梯度信息。

- 'data'表示问题的数据，它可以是任意类型的常量。当指定'data'后，以上所有函数必须增加一个输入参数来接收'data'。例如以下代码求解一个旋转的单目标优化问题：

```
d = rand(RandStream('mlfg6331_64','Seed',28),10)*2-1;
[d,~] = qr(d);
f1 = @(x,d)sum((x*d-0.5).^2);
platemo('objFcn',f1,'encoding',ones(1,10),'data',d);
```

3. 获取运行结果

算法运行结束后得到的种群可以被显示在窗口中、保存在文件中或作为函数返回值。若按以下方式调用主函数：

```
[Dec,Obj,Con] = platemo(…);
```

则最终种群会被返回，其中 Dec 表示种群的决策向量构成的矩阵、Obj 表示种群的目标值构成的矩阵、Con 表示种群的约束违反值构成的矩阵。若按以下方式调用主函数：

```
platemo('save',Value,…);
```

则当 Value 的值为负整数时（默认情况），得到的种群会被显示在窗口中，用户可以利用窗口中的 Data source 菜单来选择要显示的内容。当 Value 的值为正整数时，得到的种群会被保存在名为 PlatEMO\Data\alg\alg_pro_M_D_run.mat 的 MAT 文件中，其中 alg 表示算法名、pro 表示问题名、M 表示目标数、D 表示变量数、run 是一个自动确定的正整数以保证不和已有文件重名。每个文件存储一个单元数组 result 和一个结构体 metric，其中 result 保存得到的种群、metric 保存指标值。算法的整个优化过程被等分为 Value 块，其中 result 的第一列存储每块最后一代时所消耗的评价次数、result 的第二列存储每块最后一代时的种群、metric 存储所有种群的指标值。以上操作均由默认的输出函数@DefaultOutput 实现，用户可以通过指定'outputFcn'的值为其它函数来实现自定义的结果展示或保存方式。

```
result =
  6×2 cell array
    {[ 1650]}    {1×50 SOLUTION}
    {[ 3300]}    {1×50 SOLUTION}
    {[ 5000]}    {1×50 SOLUTION}
    {[ 6650]}    {1×50 SOLUTION}
    {[ 8300]}    {1×50 SOLUTION}
    {[10000]}    {1×50 SOLUTION}
```

```
metric =
  struct with fields:

    runtime: 0.3317
        IGD: [6×1 double]
```

此外，图形界面的实验模块可以自动计算种群的指标值并存储到 metric 中。若需要手动计算指标值，用户需载入种群、创建问题对象并调用问题的 CalMetric 方法，例如

```
% Load result
pro = DTLZ2();
pro.CalMetric('IGD',result{end});
```

其中'IGD'为要计算的指标名（参阅指标函数章节）。

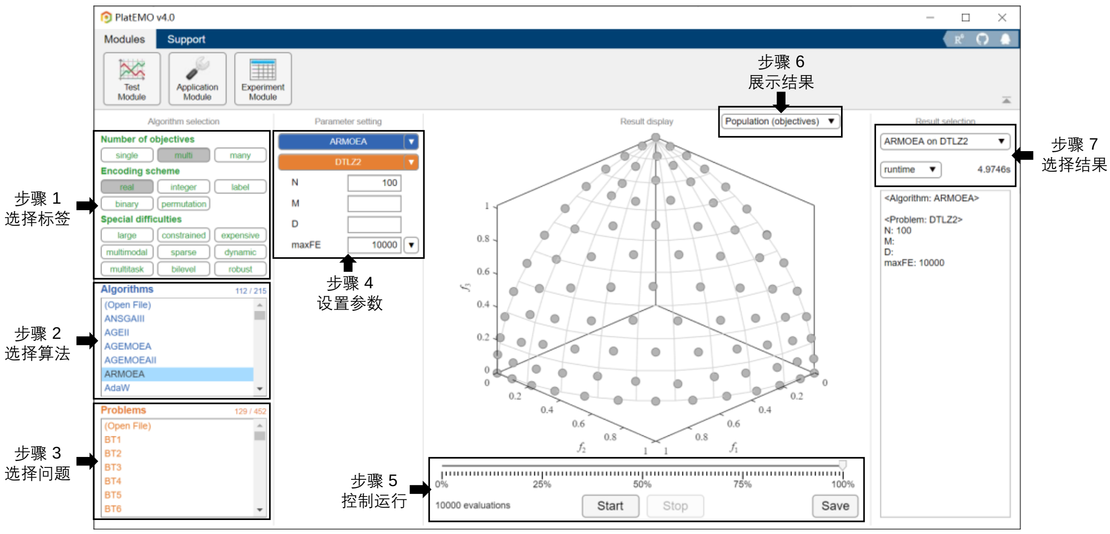
三　通过图形界面使用 PlatEMO

1. 测试模块

用户可以通过无参数调用主函数 `platemo()` 来使用 PlatEMO 的图形界面：

```
platemo();
```

图形界面的测试模块会被首先显示，它用于可视化地研究单个算法在单个问题上的性能。

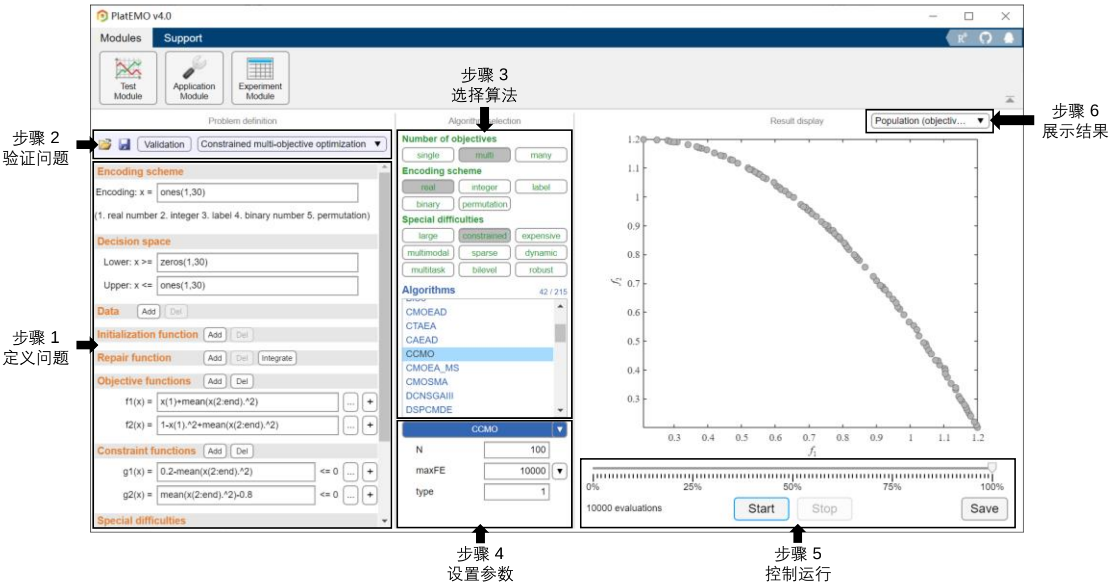


在该模块中，用户能用以下步骤研究单个算法在单个问题上的性能：

- 步骤 1：选择多个标签以确定问题类型（参阅算法、问题和指标的标签章节）。
- 步骤 2：在列表中选择一个算法。
- 步骤 3：在列表中选择一个问题。
- 步骤 4：设置算法和问题的参数。不同算法和问题可能有不同的参数，在参数上悬停可查看具体说明。
- 步骤 5：开始、暂停、停止或回退算法的运行；保存当前结果到文件。当前结果可被保存为一个 N 行 $D+M+P$ 列的矩阵，N 表示解的个数，D 表示决策变量个数，M 表示目标个数，P 表示约束个数。
- 步骤 6：选择要显示的数据，例如当前种群的目标值、变量值和各指标值。
- 步骤 7：选择要显示的历史运行结果。

2. 应用模块

用户可以通过图形界面中的菜单切换至应用模块，它用于求解自定义问题。

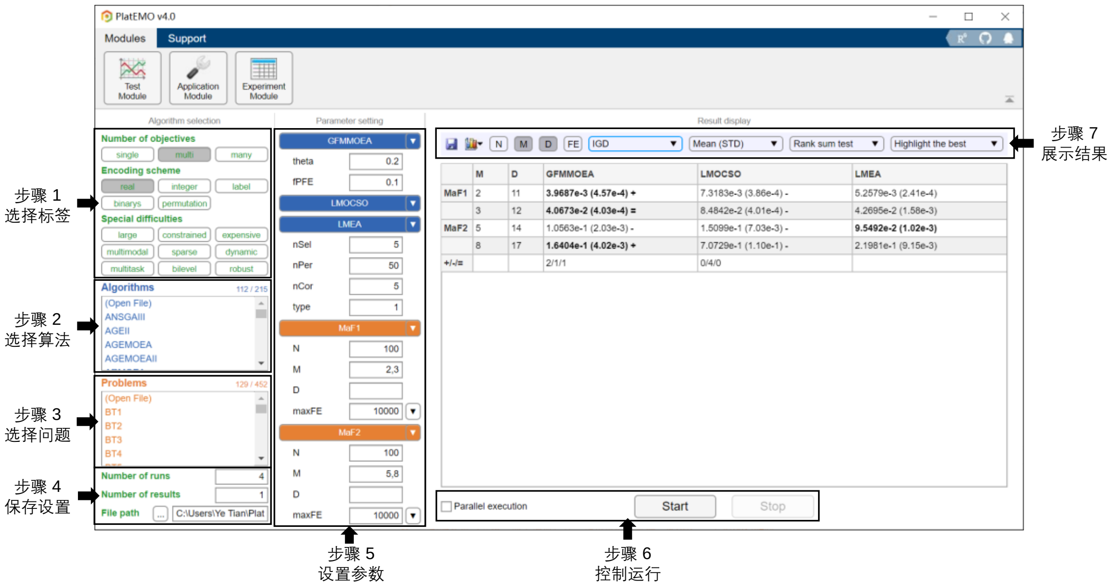


在该模块中，用户能用以下步骤求解自定义问题：

- 步骤 1：定义一个问题，定义的内容与求解自定义问题相同，其中 Encoding scheme 对应'encoding'，Decision space 对应'lower'和'upper'，Data 对应'data'，Initialization function 对应'initFcn'，Repair function 对应'decFcn'，Objective functions 对应'objFcn'，Constraint functions 对应'conFcn'，Evaluation function 对应'evalFcn'。
- 步骤 2：保存或载入问题；检测问题定义的合法性；选择一个问题模板。保存后的问题可在其它模块中打开并求解。
- 步骤 3：在列表中选择一个算法。标签会根据问题定义自动确定（参阅算法、问题和指标的标签章节）。
- 步骤 4：设置算法的参数。不同算法可能有不同的参数，在参数上悬停可查看具体说明。
- 步骤 5：开始、暂停、停止或回退算法的运行；保存当前结果到文件。当前结果可被保存为一个 N 行 $D+M+P$ 列的矩阵，N 表示解的个数，D 表示决策变量个数，M 表示目标个数，P 表示约束个数。
- 步骤 6：选择要显示的数据，例如种群的目标值、变量值和各指标值。

3. 实验模块

用户可以通过图形界面中的菜单切换至实验模块，它用于统计分析多个算法在多个问题上的性能。



在该模块中，用户能用以下步骤分析多个算法在多个问题上的性能：

- 步骤 1：选择多个标签以确定问题类型（参阅算法、问题和指标的标签章节）。
- 步骤 2：在列表中选择多个算法。
- 步骤 3：在列表中选择多个问题。
- 步骤 4：设置实验重复次数、每次保存的种群个数及保存的文件路径（参阅获取运行结果章节）。
- 步骤 5：设置算法和问题的参数。不同算法和问题可能有不同的参数，在参数上悬停可查看具体说明。
- 步骤 6：开始或停止实验的运行；选择串行（单 CPU）或并行（多 CPU）运行实验。
- 步骤 7：选择要显示的指标值；选择要执行的统计分析；保存表格到文件；将选中的多个单元格的数据显示在图窗中。

4. 算法、问题和指标的标签

每个算法、测试问题和指标需要被添加上标签，这些标签以注释的形式添加在主函数代码的第二行。例如在 PSO.m 代码的开头部分：

```
classdef PSO < ALGORITHM
```

```
% <single> <real/integer> <large/none> <constrained/none>
```

通过多个标签指定了该算法可求解的问题类型。所有的标签列举如下：

标签	描述
<single>	单目标优化：问题含有一个目标函数
<multi>	多目标优化：问题含有两或三个目标函数
<many>	超多目标优化：问题含有三个以上目标函数
<real>	连续优化：决策变量为实数
<integer>	整数优化：决策变量为整数
<label>	标签优化：决策变量为标签
<binary>	二进制优化：决策变量为二进制数
<permutation>	序列优化：决策变量构成一个全排列
<large>	大规模优化：问题含有 100 或更多的决策变量
<constrained>	约束优化：问题含有至少一个约束
<expensive>	昂贵优化：目标函数的计算非常耗时，即最大评价次数非常小
<multimodal>	多模优化：存在多个目标值接近但决策向量差异很大的最优解，它们都需要被找到
<sparse>	稀疏优化：最优解中大部分的决策变量均为零
<dynamic>	动态优化：目标函数和约束函数随时间变化
<multitask>	多任务优化：同时优化多个问题，每个问题可能含有多个目标函数和约束函数
<bilevel>	双层优化：旨在寻找上层问题的可行且最优的解，一个解对于上层问题是可行的当且仅当它是下层问题的最优解
<robust>	鲁棒优化：目标函数和约束函数受噪声影响，旨在寻找受噪声影响尽可能小且尽可能优的解
<none>	空标签
<min>	（仅用于指标）该指标值越小表示性能越好
<max>	（仅用于指标）该指标值越大表示性能越好

每个算法可能含有多个标签集合，这些集合的笛卡尔积构成该算法可求解的所有的问题类型。例如当标签集合为`<single> <real> <constrained/none>`时，表示该算法可求解带或不带约束的单目标连续优化问题；若标签集合为`<single> <real>`，表示该算法只能求解无约束问题；若标签集合为`<single> <real> <constrained>`，表示该算法只能求解有约束问题；若标签集合为`<single> <real/binary>`，表示该算法可以求解连续或二进制优化问题。

每个算法、测试问题和指标都需要被添加至少一个标签，否则它将不会在图

形界面的列表中出现。当用户在图形界面中选择多个标签后，仅有符合该标签组合的算法、测试问题和指标才会被显示供选择。标签过滤的具体原理可参阅这里。PlatEMO 中所有算法和测试问题的标签分别参阅算法列表和问题列表章节。

四　扩展 PlatEMO

1. 算法类

每个算法需要被定义为 ALGORITHM 类的子类并保存在 PlatEMO\Algorithms 文件夹中。算法类包含的属性与方法如下：

属性	赋值方式	描述
parameter	用户	算法的参数
save	用户	每次运行中保存的种群数
outputFcn	用户	在 NotTerminated() 中调用的函数
pro	Solve()	当前运行中求解的问题对象
result	NotTerminated()	当前运行中保存的种群
metric	NotTerminated()	当前保存的种群的指标值
方法	是否可重定义	描述
ALGORITHM	不可	设定由用户指定的属性值 输入：形如 'Name',Value,… 的参数设置 输出：ALGORITHM 对象
Solve	不可	利用算法求解一个问题 输入：PROBLEM 对象 输出：无
main	必须	算法的主体部分 输入：PROBLEM 对象 输出：无
NotTerminated	不可	main() 中每次迭代前调用的函数 输入：SOLUTION 对象数组，即种群 输出：是否达到终止条件（逻辑变量）
ParameterSet	不可	根据 parameter 设定算法参数 输入：默认的参数设置 输出：用户指定的参数设置

每个算法需要继承 ALGORITHM 类并重定义方法 main()。例如 GA.m 的代码为：

```
classdef GA < ALGORITHM
% <single><real/integer/label/binary/permutation><large/none><constrained/none>
% Genetic algorithm
% proC --- 1 --- Probability of crossover
```

```
5  % disC --- 20 --- Distribution index of crossover
6  % proM ---  1 --- Expectation of the number of mutated variables
7  % disM --- 20 --- Distribution index of mutation
8  
9  %------------------------ Reference -----------------------
10 % J. H. Holland, Adaptation in Natural and Artificial
11 % Systems, MIT Press, 1992.
12 %----------------------------------------------------------
13 
14     methods
15         function main(Alg,Pro)
16             [proC,disC,proM,disM] = Alg.ParameterSet(1,20,1,20);
17             P = Pro.Initialization();
18             while Alg.NotTerminated(P)
19                 Q = TournamentSelection(2,Pro.N,FitnessSingle(P));
20                 O = OperatorGA(P(Q),{proC,disC,proM,disM});
21                 P = [P,O];
22                 [~,rank] = sort(FitnessSingle(P));
23                 P = P(rank(1:Pro.N));
24             end
25         end
26     end
27 end
```

各行代码的功能如下：

第 1 行：　继承 ALGORITHM 类；

第 2 行：　为算法添加标签（参阅算法、问题和指标的标签章节）；

第 3 行：　算法的全称；

第 4-7 行：　参数名 --- 默认值 --- 参数描述，将会显示在图形界面的参数设置列表中；

第 9-12 行：算法的参考文献；

第 15 行：　重定义算法主体流程的方法；

第 16 行：　获取用户指定的参数设置，其中 1,20,1,20 分别表示参数 proC, disC,proM,disM 的默认值。

第 17 行：　调用 PROBLEM 类的方法获得一个初始种群；

第 18 行：　保存当前种群并检查是否达到终止条件；若达到终止条件则通过抛出错误强行终止算法；

第 19 行：　调用公共函数实现基于二元联赛的交配池选择；

第 20 行： 调用公共函数产生子代种群；

第 21 行： 将父子代种群合并；

第 22 行： 调用公共函数计算种群中解的适应度，并依此对解进行排序；

第 23 行： 保留适应度较好的一半解进入下一代。

在以上代码中，函数 ParameterSet() 和 NotTerminated() 是 ALGORITHM 类的方法，函数 Initialization() 是 PROBLEM 类的方法，而函数 TournamentSelection()、FitnessSingle() 和 OperatorGA() 是在 PlatEMO\Algorithms\Utility functions 文件夹中的公共函数。所有可被算法调用的方法及公共函数列举如下，详细的调用方式参阅代码中的注释。此外，函数中用于提升算法效率的技术参阅这里。

函数名	描述
ALGORITHM.NotTerminated	算法每代前调用的函数，用于保存当前种群及判断是否终止
ALGORITHM.ParameterSet	根据用户的输入设定算法参数
PROBLEM.Initialization	初始化一个种群
PROBLEM.Evaluation	评价一个种群并产生 SOLUTION 对象数组
CrowdingDistance	计算解的拥挤距离（仅用于多目标优化）
FitnessSingle	计算解的适应度（仅用于单目标优化）
NDSort	非支配排序（仅用于多目标优化）
OperatorDE	差分进化算子
OperatorFEP	进化规划算子
OperatorGA	遗传算子
OperatorGAhalf	遗传算子（仅返回前一半的子代）
OperatorPSO	粒子群优化算子
RouletteWheelSelection	轮盘赌选择
TournamentSelection	联赛选择
UniformPoint	产生均匀分布的参考点

2. 问题类

每个问题需要被定义为 PROBLEM 类的子类并保存在 PlatEMO\Problems 文件夹中。问题类包含的属性与方法如下：

属性	赋值方式	描述
N	用户	求解该问题的算法的种群大小
M	用户和 Setting()	问题的目标数
D	用户和 Setting()	问题的变量数
maxFE	用户	求解该问题可使用的最大评价次数
FE	Evaluation()	当前运行中已消耗的评价次数
maxRuntime	用户	求解该问题可使用的最大运行时间（秒）
encoding	Setting()	每个变量的编码方式
lower	Setting()	每个变量的下界
upper	Setting()	每个变量的上界
optimum	GetOptimum()	问题的最优值，例如目标函数的最小值（单目标优化）和前沿面上一组均匀参考点（多目标优化）
PF	GetPF()	问题的前沿面，例如 1 维曲线（双目标优化）、2 维曲面（三目标优化）和可行区域（约束优化）
parameter	用户	问题的参数
方法	是否可重定义	描述
PROBLEM	不可	设定由用户指定的属性值 输入：形如 'Name',Value,… 的参数设置 输出：PROBLEM 对象
Setting	必须	设定默认的属性值 输入：无 输出：无
Initialization	可以	初始化一个种群 输入：种群大小 输出：SOLUTION 对象数组，即种群
Evaluation	可以	评价一个种群并产生解对象 输入：种群的决策向量构成的矩阵 输出：SOLUTION 对象数组，即种群
CalDec	可以	修复一个种群中的无效解 输入：种群的决策向量构成的矩阵 输出：修复后的决策向量构成的矩阵
CalObj	必须	计算一个种群中解的目标值；所有目标函数均被最小化 输入：种群的决策向量构成的矩阵 输出：种群的目标值构成的矩阵
CalCon	可以	计算一个种群中解的约束违反值；当且仅当约束

		违反值小于等于零时，约束被满足 输入：种群的决策向量构成的矩阵 输出：种群的约束违反值构成的矩阵
CalObjGrad	可以	计算一个解在目标上的梯度 输入：一个决策向量 输出：雅可比矩阵
CalConGrad	可以	计算一个解在约束上的梯度 输入：一个决策向量 输出：雅可比矩阵
GetOptimum	可以	产生问题的最优值并保存在 optimum 中 输入：最优值的个数 输出：最优值集合（矩阵）
GetPF	可以	产生问题的前沿面并保存在 PF 中 输入：无 输出：用于绘制前沿面的数据（矩阵或单元数组）
CalMetric	可以	计算种群的指标值 输入一：指标名 输入二：SOLUTION 对象数组，即种群 输出：指标值（标量）
DrawDec	可以	显示一个种群的决策向量 输入：SOLUTION 对象数组，即种群 输出：无
DrawObj	可以	显示一个种群的目标向量 输入：SOLUTION 对象数组，即种群 输出：无
ParameterSet	不可	根据 parameter 设定问题参数 输入：默认的参数设置 输出：用户指定的参数设置

每个算法需要继承 PROBLEM 类并重定义方法 Setting()和 CalObj()。例如 SOP_F1.m 的代码为：

```
classdef SOP_F1 < PROBLEM
% <single><real><expensive/none>
% Sphere function

%------------------------ Reference -----------------------
% X. Yao, Y. Liu, and G. Lin, Evolutionary programming made
% faster, IEEE Transactions on Evolutionary Computation,
% 1999, 3(2): 82-102.
%----------------------------------------------------------

```

```
11      methods
12          function Setting(obj)
13              obj.M = 1;
14              if isempty(obj.D); obj.D = 30; end
15              obj.lower = zeros(1,obj.D) - 100;
16              obj.upper = zeros(1,obj.D) + 100;
17              obj.encoding = ones(1,obj.D);
18          end
19          function PopObj = CalObj(obj,PopDec)
20              PopObj = sum(PopDec.^2,2);
21          end
22      end
23  end
```

各行代码的功能如下：

第 1 行： 继承 PROBLEM 类；

第 2 行： 为问题添加标签（参阅算法、问题和指标的标签章节）；

第 3 行： 问题的全称；

第 5-9 行： 问题的参考文献；

第 12 行： 重定义设定默认属性值的方法；

第 13 行： 设置问题的目标数；

第 14 行： 设置问题的变量数（若未被用户指定）；

第15-16行：设置决策变量的上下界；

第 17 行： 设置决策变量的编码方式；

第 19 行： 重定义计算目标函数的方法；

第 20 行： 计算种群中解的目标值。

除以上代码外，默认的方法 Initialization()用于随机初始化一个种群，用户可以重定义该方法来指定特殊的种群初始化策略。例如 Sparse_NN.m 将初始化的种群中随机一半的决策变量置零：

```
function Population = Initialization(obj,N)
    if nargin < 2; N = obj.N; end
    PopDec = (rand(N,obj.D)-0.5)*2.*randi([0 1],N,obj.D);
    Population = obj.Evaluation(PopDec);
end
```

默认的方法 CalDec()将大于上界的决策变量设为上界值、将小于下界的决策变量设为下界值，用户可以重定义该方法来指定特殊的解修复策略。例如 MOKP.m

修复了超过背包容量限制的解，使得该问题无需添加约束函数：

```
function PopDec = CalDec(obj,PopDec)
    C = sum(obj.W,2)/2;
    [~,rank] = sort(max(obj.P./obj.W));
    for i = 1 : size(PopDec,1)
        while any(obj.W*PopDec(i,:)'>C)
            k = find(PopDec(i,rank),1);
            PopDec(i,rank(k)) = 0;
        end
    end
end
```

默认的方法 `CalCon()` 返回零作为解的约束违反值（即解都是满足约束的），用户可以重定义该方法来指定问题的约束。例如 `CF4.m` 添加了一个约束：

```
function PopCon = CalCon(obj,X)
    t = X(:,2)-sin(6*pi*X(:,1)+2*pi/size(X,2))-0.5*X(:,1)+0.25;
    PopCon = -t./(1+exp(4*abs(t)));
end
```

利用 `all(PopCon<=0,2)` 可确定每个解是否满足所有约束。注意等式约束必须转换为不等式约束来处理。默认的方法 `Evaluation()` 通过依次调用 `CalDec()`、`CalObj()` 和 `CalCon()` 来实例化 `SOLUTION` 对象，同时增加已消耗评价次数 `FE` 的值。用户可以重定义该方法来在一个函数内完成种群的修复、目标计算和约束计算的工作，此时 `CalDec()`、`CalObj()` 和 `CalCon()` 将不会被调用。例如 `MW2.m` 同时计算了种群的目标值与约束违反值：

```
function Population = Evaluation(obj,varargin)
    X = varargin{1};
    X=max(min(X,repmat(obj.upper,size(X,1),1)),repmat(obj.lower,size(X,1),1));
    z=1-exp(-10*(X(:,obj.M:end)-(repmat(obj.M:obj.D,size(X,1),1)-1)/obj.D).^2);
    g = 1+sum((1.5+(0.1/obj.D)*z.^2-1.5*cos(2*pi*z)),2);
    PopObj(:,1) = X(:,1);
    PopObj(:,2) = g.*(1-PopObj(:,1)./g);
    L = sqrt(2)*PopObj(:,2)-sqrt(2)*PopObj(:,1);
    PopCon = sum(PopObj,2)-1-0.5*sin(3*pi*l).^8;
    Population = SOLUTION(X,PopObj,PopCon,varargin{2:end});
    obj.FE = obj.FE+length(Population);
end
```

默认的方法 `CalObjGrad()` 通过有限差分来估计目标函数的梯度，用户可以重

定义该方法以更准确地计算梯度。类似地，默认的方法 CalConGrad()通过有限差分来估计约束函数的梯度，用户可以重定义该方法以更准确地计算梯度。用户可以重定义方法 GetOptimum()来指定问题的最优值，最优值被用于指标值的计算。例如 SOP_F8.m 指定了目标函数的最小值：

```
function R = GetOptimum(obj,N)
    R = -418.9829*obj.D;
end
```

DTLZ2.m 生成了一组前沿面上均匀分布的参考点：

```
function R = GetOptimum(obj,N)
    R = UniformPoint(N,obj.M);
    R = R./repmat(sqrt(sum(R.^2,2)),1,obj.M);
end
```

在不同形状前沿面上的采点方法参阅这里。用户可以重定义方法 GetPF()来指定多目标优化问题的前沿面或可行区域，它们被用于 DrawObj()的可视化中。例如 DTLZ2.m 生成了 2 维和 3 维的前沿面数据：

```
function R = GetPF(obj)
    if obj.M == 2
        R = obj.GetOptimum(100);
    elseif obj.M == 3
        a = linspace(0,pi/2,10)';
        R = {sin(a)*cos(a'),sin(a)*sin(a'),cos(a)*ones(size(a'))};
    else
        R = [];
    end
end
```

MW1.m 生成了可行区域的数据：

```
function R = GetPF(obj)
    [x,y]= meshgrid(linspace(0,1,400),linspace(0,1.5,400));
    z     = nan(size(x));
    fes   = x+y-1-0.5*sin(2*pi*(sqrt(2)*y-sqrt(2)*x)).^8 <= 0;
    z(fes&0.85*x+y>=1) = 0;
    R     = {x,y,z};
end
```

默认的方法 CalMetric()将一个种群与问题的最优值 optimum 传入指标函数中进行计算，用户可以重定义该方法来将不同的变量传入指标函数中。例如

SMMOP1.m 在计算 IGDX 指标时传入问题的最优解集而非前沿面上的参考点：

```
function score = CalMetric(obj,metName,Population)
    switch metName
        case 'IGDX'
            score = feval(metName,Population,obj.POS);
        otherwise
            score = feval(metName,Population,obj.optimum);
    end
end
```

默认的方法 DrawDec()显示种群的决策向量（用于图形界面中），用户可以重定义该方法来指定特殊的显示方式。例如 TSP.m 显示了种群中最优解的路径：

```
function DrawDec(obj,P)
    [~,best] = min(P.objs);
    Draw(obj.R(P(best).dec([1:end,1]),:),'-k','LineWidth',1.5);
    Draw(obj.R);
end
```

默认的方法 DrawObj()显示种群的目标向量（用于图形界面中），用户可以重定义该方法来指定特殊的显示方式。例如 Sparse_CD.m 添加了坐标轴的标签：

```
function DrawObj(obj,P)
    Draw(P.objs,{'Kernel k-means','Ratio cut',[]});
end
```

其中 Draw()用于显示数据，它位于 PlatEMO\GUI 文件夹中。

3. 个体类

一个 SOLUTION 类的对象表示一个个体（即一个解），一组 SOLUTION 类的对象表示一个种群。个体类包含的属性与方法如下：

属性	赋值方式	描述
dec	PROBLEM.Evaluation()	解的决策向量
obj	PROBLEM.Evaluation()	解的目标值
con	PROBLEM.Evaluation()	解的约束违反值
add	PROBLEM.Evaluation()	解的额外属性值（例如速度）
方法	描述	

SOLUTION	生成 SOLUTION 对象数组 输入一：多个解的决策向量构成的矩阵 输入二：多个解的目标值构成的矩阵 输入三：多个解的约束违反值构成的矩阵 输入四：多个解的额外属性值构成的矩阵 输出：SOLUTION 对象数组
decs	获取多个解的决策向量 输入：无 输出：多个解的决策向量构成的矩阵
objs	获取多个解的目标值 输入：无 输出：多个解的目标值构成的矩阵
cons	获取多个解的约束违反值 输入：无 输出：多个解的约束违反值构成的矩阵
adds	设置并获取多个解的额外属性值 输入：默认的额外属性值 输出：多个解的额外属性值构成的矩阵
best	获取种群中可行且最好的解（单目标优化）或可行且非支配的解（多目标优化） 输入：无 输出：种群中可行且最好的 SOLUTION 对象子数组

例如，以下代码产生一个具有十个解的种群，并获取其中最好的解的目标值矩阵：

```
Population = SOLUTION(rand(10,5),rand(10,1),zeros(10,1));
BestObjs   = Population.best.objs
```

注意在 SOLUTION()只能在 PROBLEM 类的 Evaluation()方法内调用。

4. 一次完整的运行过程

以下代码利用遗传算法求球面函数的最小值：

```
Alg = GA();
Pro = SOP_F1();
Alg.Solve(Pro);
```

其中代码 Alg.Solve(Pro)执行时所涉及的函数调用过程如下图所示。

5. 指标函数

每个性能指标需要被定义为一个函数并保存在 PlatEMO\Metrics 文件夹中。例如 IGD.m 的代码为：

```
function score = IGD(Population,optimum)
% <min> <multi/many> <real/integer/label/binary/permutation> <large/none> <constrained/none> <expensive/none> <multimodal/none> <sparse/none> <dynamic/none> <robust/none>
% Inverted generational distance

%------------------------ Reference -----------------------
% C. A. Coello Coello and N. C. Cortes, Solving
% multiobjective optimization problem using an artificial
% immune system, Genetic Programming and Evolvable
```

```
9  % Machines, 2005, 6(2): 163-190.
10 %------------------------------------------------------------
11 
12     PopObj = Population.best.objs;
13     if size(PopObj,2) ~= size(optimum,2)
14         score = nan;
15     else
16         score = mean(min(pdist2(optimum,PopObj),[],2));
17     end
18 end
```

各行代码的功能如下：

第 1 行： 函数声明，其中第一个输入为一个种群（即一个 SOLUTION 对象数组）、第二个输入为问题的最优值（即问题的 optimum 属性）、输出为种群的指标值；

第 2 行： 为指标添加标签（参阅算法、问题和指标的标签章节）；注意标签 <min>或<max>必须为第一个标签；

第 3 行： 指标的全称；

第 5-10 行：指标的参考文献；

第 12 行： 获取种群中最好的解（可行且非支配的解）的目标值矩阵；

第13-14行：若种群不存在可行解则返回 nan；

第15-16行：否则返回可行且非支配的解的指标值。

五 算法列表

	算法缩写	算法全称	single	multi	many	real	integer	label	binary	permutation	large	constrained	expensive	multimodal	sparse	dynamic	multitask	bilevel	robust
1	ABC	Artificial bee colony algorithm	√			√	√				√	√							
2	AB-SAEA	Adaptive Bayesian based surrogate-assisted evolutionary algorithm		√	√	√	√						√						
3	ACO	Ant colony optimization	√							√	√								
4	Adam	Adaptive moment estimation	√			√					√								
5	AdaW	Evolutionary algorithm with adaptive weights		√	√	√	√	√	√	√									
6	AGE-II	Approximation-guided evolutionary multi-objective algorithm II		√		√	√	√	√	√									
7	AGE-MOEA	Adaptive geometry estimation-based many-objective evolutionary algorithm		√	√	√	√	√	√	√		√							
8	AGE-MOEA-II	Adaptive geometry estimation-based many-objective evolutionary algorithm II		√	√	√	√	√	√	√		√							
9	A-NSGA-III	Adaptive NSGA-III		√	√	√	√	√	√	√		√							
10	AR-MOEA	Adaptive reference points based multi-objective evolutionary algorithm		√	√	√	√	√	√	√		√							
11	BCE-IBEA	Bi-criterion evolution based IBEA		√	√	√	√	√	√	√									
12	BCE-MOEA/D	Bi-criterion evolution based MOEA/D		√	√	√	√	√	√	√									
13	BFGS	A quasi-Newton method proposed by Broyden, Fletcher, Goldfarb, and Shanno	√			√					√								
14	BiCo	Bidirectional coevolution constrained multiobjective evolutionary algorithm		√		√	√	√	√	√		√							
15	BiGE	Bi-goal evolution			√	√	√	√	√	√									
16	BLEAQII	Bilevel evolutionary algorithm based on quadratic approximations II		√		√						√						√	
17	BSPGA	Binary space partition tree based genetic algorithm	√						√		√	√							
18	CAEAD	Dual-population evolutionary algorithm based on alternative evolution and degeneration		√		√	√	√	√	√		√							
19	CA-MOEA	Clustering based adaptive multi-objective evolutionary algorithm		√		√	√	√	√	√									
20	CCGDE3	Cooperative coevolution GDE3		√		√	√				√								
21	CCMO	Coevolutionary constrained multi-objective optimization framework		√		√	√	√	√	√		√							
22	c-DPEA	Constrained dual-population evolutionary algorithm		√		√	√	√	√	√		√							
23	CLIA	Evolutionary algorithm with cascade clustering and reference point incremental learning		√	√	√	√	√	√	√									
24	CMA-ES	Covariance matrix adaptation evolution strategy	√			√	√				√	√							
25	C-MOEA/D	Constraint-MOEA/D		√	√	√	√	√	√	√		√							
26	CMOEA-MS	Constrained multiobjective evolutionary algorithm with multiple stages		√		√	√	√	√	√		√							

	算法缩写	算法全称	single	multi	many	real	integer	label	binary	permutation	large	constrained	expensive	multimodal	sparse	dynamic	multitask	bilevel	robust
27	CMOPSO	Competitive mechanism based multi-objective particle swarm optimizer		√		√	√												
28	CMOSMA	Constrained multi-objective evolutionary algorithm with self-organizing map		√	√	√	√					√							
29	CNSDE/DVC	Constrained nondominated sorting differential evolution based on decision variable classification		√		√	√												√
30	CPS-MOEA	Classification and Pareto domination based multi-objective evolutionary		√		√	√						√						
31	CSEA	Classification based surrogate-assisted evolutionary algorithm		√	√	√							√						
32	CSO	Competitive swarm optimizer	√			√	√				√	√							
33	C-TAEA	Two-archive evolutionary algorithm for constrained MOPs		√	√	√	√	√	√	√		√							
34	DAEA	Duplication analysis based evolutionary algorithm		√					√										
35	DCNSGA-III	Dynamic constrained NSGA-III		√	√	√	√	√	√	√		√							
36	DE	Differential evolution	√			√	√				√	√							
37	DEA-GNG	Decomposition based evolutionary algorithm guided by growing neural gas		√	√	√	√	√	√	√									
38	DGEA	Direction guided evolutionary algorithm		√	√	√	√				√								
39	DMOEA-eC	Decomposition-based multi-objective evolutionary algorithm with the e-constraint framework		√		√	√	√	√	√									
40	dMOPSO	MOPSO based on decomposition		√		√	√												
41	DN-NSGA-II	Decision space based niching NSGA-II		√		√	√							√					
42	DNSGA-II	Dynamic NSGA-II		√		√	√	√	√	√						√			
43	DSPCMDE	Dynamic selection preference-assisted constrained multiobjective differential evolution		√		√	√					√							
44	DWU	Dominance-weighted uniformity multi-objective evolutionary algorithm		√		√	√	√	√	√									
45	EAG-MOEA/D	External archive guided MOEA/D		√		√	√	√	√	√									
46	EDN-ARMOEA	Efficient dropout neural network based AR-MOEA		√	√	√	√						√						
47	EFR-RR	Ensemble fitness ranking with a ranking restriction scheme		√	√	√	√	√	√	√									
48	EGO	Efficient global optimization	√			√	√						√						
49	EIM-EGO	Expected improvement matrix based efficient global optimization		√		√	√						√						
50	EMCMO	Evolutionary multitasking-based constrained multiobjective optimization		√		√	√	√	√	√		√							
51	e-MOEA	Epsilon multi-objective evolutionary algorithm		√	√	√	√	√	√	√									
52	EMyO/C	Evolutionary many-objective optimization algorithm with clustering-based		√	√	√	√												
53	ENS-MOEA/D	Ensemble of different neighborhood sizes based MOEA/D		√	√	√	√												
54	FDV	Fuzzy decision variable framework with various internal optimizers		√	√	√	√				√								
55	FEP	Fast evolutionary programming	√			√	√				√	√							

	算法缩写	算法全称	single	multi	many	real	integer	label	binary	permutation	large	constrained	expensive	multimodal	sparse	dynamic	multitask	bilevel	robust
56	FRCG	Fletcher-Reeves conjugate gradient	√			√					√								
57	FRCGM	Fletcher-Reeves conjugate gradient (for multi-objective optimization)		√	√	√					√	√							
58	FROFI	Feasibility rule with the incorporation of objective function information	√			√	√				√	√							
59	GA	Genetic algorithm	√			√	√	√	√	√	√	√							
60	GDE3	Generalized differential evolution 3		√		√	√					√							
61	GFM-MOEA	Generic front modeling based multi-objective evolutionary algorithm		√	√	√	√	√	√	√									
62	GLMO	Grouped and linked mutation operator algorithm		√		√	√				√								
63	g-NSGA-II	g-dominance based NSGA-II		√		√	√	√	√	√									
64	GPSO	Gradient based particle swarm optimization algorithm	√			√					√	√							
65	GPSOM	Gradient based particle swarm optimization algorithm (for multi-objective optimization)		√	√	√					√	√							
66	GrEA	Grid-based evolutionary algorithm			√	√	√	√	√	√									
67	HeE-MOEA	Multiobjective evolutionary algorithm with heterogeneous ensemble based infill criterion		√		√	√						√						
68	hpaEA	Hyperplane assisted evolutionary algorithm		√	√	√	√	√	√	√									
69	HREA	Hierarchy ranking based evolutionary algorithm		√		√	√							√					
70	HypE	Hypervolume estimation algorithm		√	√	√	√	√	√	√									
71	IBEA	Indicator-based evolutionary algorithm		√	√	√	√	√	√	√									
72	ICMA	Indicator based constrained multi-objective algorithm		√		√	√					√							
73	I-DBEA	Improved decomposition-based evolutionary algorithm		√	√	√	√	√	√	√		√							
74	IM-MOEA	Inverse modeling based multiobjective evolutionary algorithm		√		√	√				√								
75	IM-MOEA/D	Inverse modeling multiobjective evolutionary algorithm based on decomposition		√		√	√				√								
76	IMODE	Improved multi-operator differential evolution	√			√	√				√	√							
77	I-SIBEA	Interactive simple indicator-based evolutionary algorithm		√		√	√	√	√	√									
78	Izui	An aggregative gradient based multi-objective optimizer proposed by Izui et al.		√	√	√					√	√							
79	KnEA	Knee point driven evolutionary algorithm			√	√	√	√	√	√		√							
80	K-RVEA	Surrogate-assisted RVEA		√	√	√	√						√						
81	KTA2	Kriging-assisted Two_Arch2		√	√	√	√						√						
82	LCSA	Linear combination-based search algorithm		√	√	√	√				√								
83	LMEA	Evolutionary algorithm for large-scale many-objective optimization		√	√	√	√				√								
84	LMOCSO	Large-scale multi-objective competitive swarm optimization algorithm		√	√	√	√				√	√							
85	LMOEA-DS	Large-scale evolutionary multi-objective		√		√	√				√								

	算法缩写	算法全称	single	multi	many	real	integer	label	binary	permutation	large	constrained	expensive	multimodal	sparse	dynamic	multitask	bilevel	robust
		optimization assisted by directed sampling																	
86	LMPFE	Evolutionary algorithm with local model based Pareto front estimation		√	√	√	√	√	√	√									
87	LSMOF	Large-scale multi-objective optimization framework with NSGA-II		√		√	√				√								
88	MaOEA-CSS	Many-objective evolutionary algorithms based on coordinated selection		√	√	√	√	√	√	√									
89	MaOEA-DDFC	Many-objective evolutionary algorithm based on directional diversity and favorable convergence		√	√	√	√	√	√	√									
90	MaOEA/IGD	IGD based many-objective evolutionary algorithm			√	√	√	√	√	√									
91	MaOEA/IT	Many-objective evolutionary algorithms based on an independent two-stage		√	√	√	√					√							
92	MaOEA-R&D	Many-objective evolutionary algorithm based on objective space reduction			√	√	√	√	√	√									
93	MCEA/D	Multiple classifiers-assisted evolutionary algorithm based on decomposition		√	√	√	√						√						
94	MFEA	Multifactorial evolutionary algorithm	√			√	√	√	√	√	√						√		
95	MFEA-II	Multifactorial evolutionary algorithm II	√			√	√	√	√	√	√						√		
96	MMEA-WI	Weighted indicator-based evolutionary algorithm for multimodal multi-objective optimization		√		√	√							√					
97	MMOPSO	MOPSO with multiple search strategies		√		√	√												
98	MO_Ring_PSO_SCD	Multiobjective PSO using ring topology and special crowding distance		√		√	√							√					
99	MOCell	Cellular genetic algorithm		√		√	√	√	√	√		√							
100	MOCGDE	Multi-objective conjugate gradient and differential evolution algorithm		√	√	√					√	√							
101	MO-CMA	Multi-objective covariance matrix adaptation evolution strategy		√		√	√												
102	MOEA/D	Multiobjective evolutionary algorithm based on decomposition		√	√	√	√	√	√	√									
103	MOEA/D-AWA	MOEA/D with adaptive weight adjustment		√	√	√	√	√	√	√									
104	MOEA/D-CMA	MOEA/D with covariance matrix adaptation evolution strategy		√	√	√	√												
105	MOEA/DD	Many-objective evolutionary algorithm based on dominance and decomposition		√	√	√	√	√	√	√		√							
106	MOEA/D-DAE	MOEA/D with detect-and-escape strategy		√		√	√	√	√	√		√							
107	MOEA/D-DCWV	MOEA/D with distribution control of weight vector set		√	√	√	√	√	√	√									
108	MOEA/D-DE	MOEA/D based on differential evolution		√	√	√	√												
109	MOEA/D-DRA	MOEA/D with dynamical resource allocation		√	√	√	√												
110	MOEA/D-DU	MOEA/D with a distance based updating strategy		√	√	√	√	√	√	√									
111	MOEA/D-DYTS	MOEA/D with dynamic Thompson sampling		√	√	√	√												
112	MOEA/D-EGO	MOEA/D with efficient global optimization		√		√	√						√						
113	MOEA/D-FRRMAB	MOEA/D with fitness-rate-rank-based multiarmed bandit		√	√	√	√												

	算法缩写	算法全称	single	multi	many	real	integer	label	binary	permutation	large	constrained	expensive	multimodal	sparse	dynamic	multitask	bilevel	robust
114	MOEA/D-M2M	MOEA/D based on MOP to MOP		√		√	√												
115	MOEA/D-MRDL	MOEA/D with maximum relative diversity loss		√		√	√												
116	MOEA/D-PaS	MOEA/D with Pareto adaptive scalarizing approximation		√	√	√	√												
117	MOEA/D-PFE	MOEA/D with Pareto front estimation		√	√	√	√	√	√	√									
118	MOEA/D-STM	MOEA/D with stable matching		√	√	√	√												
119	MOEA/D-UR	MOEA/D with update when required		√	√	√	√	√	√	√									
120	MOEA/D-URAW	MOEA/D with uniform randomly adaptive weights		√	√	√	√	√	√	√									
121	MOEA/DVA	Multi-objective evolutionary algorithm based on decision variable		√		√	√				√								
122	MOEA/D-VOV	MOEA/D with virtual objective vectors		√	√	√	√	√	√	√									
123	MOEA/IGD-NS	Multi-objective evolutionary algorithm based on an enhanced IGD		√		√	√	√	√	√									
124	MOEA-PC	Multiobjective evolutionary algorithm based on polar coordinates		√		√	√												
125	MOEA/PSL	Multi-objective evolutionary algorithm based on Pareto optimal subspace		√		√	√		√		√	√			√				
126	MOEA-RE	Multi-objective evolutionary algorithm with robustness enhancement		√		√	√	√	√	√									√
127	MO-EGS	Multi-objective evolutionary gradient search		√		√					√								
128	MOMBI-II	Many objective metaheuristic based on the R2 indicator II		√	√	√	√	√	√	√									
129	MO-MFEA	Multi-objective multifactorial evolutionary algorithm		√		√	√	√	√	√		√					√		
130	MO-MFEA-II	Multi-objective multifactorial evolutionary algorithm II		√		√	√	√	√	√		√					√		
131	MOPSO	Multi-objective particle swarm optimization		√		√	√												
132	MOPSO-CD	MOPSO with crowding distance		√		√	√												
133	MOSD	Multiobjective steepest descent		√		√					√	√							
134	M-PAES	Memetic algorithm with Pareto archived evolution strategy		√		√	√												
135	MP-MMEA	Multi-population multi-modal multi-objective evolutionary algorithm		√		√	√				√			√	√				
136	MPSO/D	Multi-objective particle swarm optimization algorithm based on decomposition		√	√	√	√												
137	MSCMO	Multi-stage constrained multi-objective evolutionary algorithm		√		√	√	√	√	√		√							
138	MSEA	Multi-stage multi-objective evolutionary algorithm		√		√	√	√	√	√									
139	MSKEA	Multi-stage knowledge-guided evolutionary algorithm		√		√	√		√		√	√			√				
140	MSOPS-II	Multiple single objective Pareto sampling II		√	√	√	√					√							
141	MTCMO	Multitasking constrained multi-objective optimization		√		√	√	√	√	√		√							

	算法缩写	算法全称	single	multi	many	real	integer	label	binary	permutation	large	constrained	expensive	multimodal	sparse	dynamic	multitask	bilevel	robust
142	MTS	Multiple trajectory search		√		√	√												
143	MultiObjective EGO	Multi-objective efficient global optimization		√		√	√					√	√						
144	MyO-DEMR	Many-objective differential evolution with mutation restriction		√	√	√	√												
145	NBLEA	Nested bilevel evolutionary algorithm		√		√						√						√	
146	NelderMead	The Nelder-Mead algorithm	√			√													
147	NMPSO	Novel multi-objective particle swarm optimization		√	√	√	√												
148	NNIA	Nondominated neighbor immune algorithm		√		√	√	√	√	√									
149	NSGA-II	Nondominated sorting genetic algorithm II		√		√	√	√	√	√		√							
150	NSGA-II+ARSBX	NSGA-II with adaptive rotation based simulated binary crossover		√		√	√					√							
151	NSGA-II-conflict	NSGA-II with conflict-based partitioning strategy			√	√	√	√	√	√									
152	NSGA-II-DTI	NSGA-II of Deb's type I robust version		√		√	√	√	√	√		√							√
153	NSGA-III	Nondominated sorting genetic algorithm III		√	√	√	√	√	√	√		√							
154	NSGA-II/SDR	NSGA-II with strengthened dominance relation			√	√	√	√	√	√									
155	NSLS	Multiobjective optimization framework based on nondominated sorting and local search		√		√	√												
156	OFA	Optimal foraging algorithm	√			√	√				√	√							
157	one-by-one EA	Many-objective evolutionary algorithm using a one-by-one selection		√	√	√	√	√	√	√									
158	OSP-NSDE	Non-dominated sorting differential evolution with prediction in the objective space		√		√	√												
159	ParEGO	Efficient global optimization for Pareto optimization		√		√	√						√						
160	PB-NSGA-III	NSGA-III based on Pareto based bi-indicator infill sampling criterion		√	√	√	√						√						
161	PB-RVEA	RVEA based on Pareto based bi-indicator infill sampling criterion		√	√	√	√						√						
162	PeEA	Pareto front shape estimation based evolutionary algorithm		√	√	√	√	√	√	√									
163	PESA-II	Pareto envelope-based selection algorithm II		√		√	√	√	√	√									
164	PICEA-g	Preference-inspired coevolutionary algorithm with goals		√	√	√	√	√	√	√									
165	PM-MOEA	Pattern mining based multi-objective evolutionary algorithm		√		√	√		√		√	√			√				
166	POCEA	Paired offspring generation based constrained evolutionary algorithm		√		√	√				√	√							
167	PPS	Push and pull search algorithm		√	√	√	√					√							
168	PREA	Promising-region based EMO algorithm		√	√	√	√	√	√	√									
169	PSO	Particle swarm optimization	√			√	√				√	√							
170	REMO	Expensive multiobjective optimization by relation learning and prediction		√	√	√							√						
171	RM-MEDA	Regularity model-based multiobjective estimation of distribution		√		√	√												

	算法缩写	算法全称	single	multi	many	real	integer	label	binary	permutation	large	constrained	expensive	multimodal	sparse	dynamic	multitask	bilevel	robust
172	RMOEA/DVA	Robust multi-objective evolutionary algorithm with decision variable assortment		√		√	√												√
173	RMSProp	Root mean square propagation	√			√					√								
174	r-NSGA-II	r-dominance based NSGA-II		√		√	√	√	√	√									
175	RPD-NSGA-II	Reference point dominance-based NSGA-II		√	√	√	√	√	√	√									
176	RPEA	Reference points-based evolutionary algorithm			√	√	√	√	√	√									
177	RSEA	Radial space division based evolutionary algorithm		√	√	√	√	√	√	√									
178	RVEA	Reference vector guided evolutionary algorithm		√	√	√	√	√	√	√		√							
179	RVEAa	RVEA embedded with the reference vector regeneration strategy			√	√	√	√	√	√									
180	RVEA-iGNG	RVEA based on improved growing neural gas		√	√	√	√	√	√	√									
181	S3-CMA-ES	Scalable small subpopulations based covariance matrix adaptation		√	√	√	√				√								
182	SA	Simulated annealing	√			√	√				√	√							
183	SACC-EAM-II	Surrogate-assisted cooperative co-evolutionary algorithm of Minamo	√			√	√						√						
184	SACOSO	Surrogate-assisted cooperative swarm optimization	√			√	√				√		√						
185	SADE-Sammon	Sammon mapping assisted differential evolution	√			√	√						√						
186	SAMSO	Multiswarm-assisted expensive optimization	√			√	√				√		√						
187	S-CDAS	Self-controlling dominance area of solutions			√	√	√	√	√	√									
188	SD	Steepest descent	√			√					√								
189	S-ECSO	Enhanced competitive swarm optimizer for sparse optimization		√		√					√				√				
190	SGEA	Steady-state and generational evolutionary algorithm		√		√	√	√	√	√		√				√			
191	SHADE	Success-history based adaptive differential evolution	√			√	√				√	√							
192	SIBEA	Simple indicator-based evolutionary algorithm		√		√	√	√	√	√									
193	SIBEA-kEMOSS	SIBEA with minimum objective subset of size k with minimum error			√	√	√	√	√	√									
194	SLMEA	Super-large-scale multi-objective evolutionary algorithm		√		√	√		√		√	√			√				
195	SMEA	Self-organizing multiobjective evolutionary algorithm		√		√	√												
196	SMPSO	Speed-constrained multi-objective particle swarm optimization		√		√	√												
197	SMS-EGO	S metric selection based efficient global optimization		√		√	√						√						
198	SMS-EMOA	S metric selection based evolutionary multiobjective optimization		√		√	√	√	√	√									
199	SparseEA	Evolutionary algorithm for sparse multi-objective optimization problems		√		√	√		√		√	√			√				
200	SparseEA2	Improved SparseEA		√		√	√		√		√	√			√				
201	SPEA2	Strength Pareto evolutionary algorithm 2		√		√	√	√	√	√									

	算法缩写	算法全称	single	multi	many	real	integer	label	binary	permutation	large	constrained	expensive	multimodal	sparse	dynamic	multitask	bilevel	robust
202	SPEA2+SDE	SPEA2 with shift-based density estimation			√	√	√	√	√	√									
203	SPEA/R	Strength Pareto evolutionary algorithm based on reference direction		√	√	√	√	√	√	√									
204	SQP	Sequential quadratic programming	√			√					√	√							
205	SRA	Stochastic ranking algorithm			√	√	√	√	√	√									
206	t-DEA	theta-dominance based evolutionary algorithm		√	√	√	√	√	√	√									
207	TiGE-2	Tri-Goal Evolution Framework for CMaOPs			√	√	√	√	√	√		√							
208	ToP	Two-phase framework with NSGA-II		√		√	√					√							
209	TriMOEA-TA&R	Multi-modal MOEA using two-archive and recombination strategies		√		√	√							√					
210	TriP	Tri-population based coevolutionary algorithm		√	√	√	√					√							
211	TSTI	Two-stage evolutionary algorithm with three indicators		√		√	√	√	√	√		√							
212	Two_Arch2	Two-archive algorithm 2		√	√	√	√	√	√	√									
213	URCMO	Utilizing the relationship between constrained and unconstrained Pareto fronts for constrained multi-objective optimization		√		√	√					√							
214	VaEA	Vector angle based evolutionary algorithm		√	√	√	√	√	√	√									
215	WOF	Weighted optimization framework		√		√	√				√								
216	WV-MOEA-P	Weight vector based multi-objective optimization algorithm with preference		√		√	√												

六 问题列表

	问题缩写	问题全称	single	multi	many	real	integer	label	binary	permutation	large	constrained	expensive	multimodal	sparse	dynamic	multitask	bilevel	robust
1	BT1	Benchmark MOP with bias feature		√		√					√								
2	BT2	Benchmark MOP with bias feature		√		√					√								
3	BT3	Benchmark MOP with bias feature		√		√					√								
4	BT4	Benchmark MOP with bias feature		√		√					√								
5	BT5	Benchmark MOP with bias feature		√		√					√								
6	BT6	Benchmark MOP with bias feature		√		√					√								
7	BT7	Benchmark MOP with bias feature		√		√					√								
8	BT8	Benchmark MOP with bias feature		√		√					√								
9	BT9	Benchmark MOP with bias feature		√		√					√								
10	CEC2008_F1	Shifted sphere function	√			√					√		√						
11	CEC2008_F2	Shifted Schwefel's function	√			√					√		√						
12	CEC2008_F3	Shifted Rosenbrock's function	√			√					√		√						
13	CEC2008_F4	Shifted Rastrign's function	√			√					√		√						
14	CEC2008_F5	Shifted Griewank's function	√			√					√		√						
15	CEC2008_F6	Shifted Ackley's function	√			√					√		√						
16	CEC2008_F7	FastFractal 'DoubleDip' function	√			√					√		√						
17	CEC2010_F1	CEC'2010 constrained optimization benchmark problem	√			√						√							
18	CEC2010_F2	CEC'2010 constrained optimization benchmark problem	√			√						√							
19	CEC2010_F3	CEC'2010 constrained optimization benchmark problem	√			√						√							
20	CEC2010_F4	CEC'2010 constrained optimization benchmark problem	√			√						√							
21	CEC2010_F5	CEC'2010 constrained optimization benchmark problem	√			√						√							
22	CEC2010_F6	CEC'2010 constrained optimization benchmark problem	√			√						√							
23	CEC2010_F7	CEC'2010 constrained optimization benchmark problem	√			√						√							
24	CEC2010_F8	CEC'2010 constrained optimization benchmark problem	√			√						√							
25	CEC2010_F9	CEC'2010 constrained optimization benchmark problem	√			√						√							
26	CEC2010_F10	CEC'2010 constrained optimization benchmark problem	√			√						√							
27	CEC2010_F11	CEC'2010 constrained optimization benchmark problem	√			√						√							

	问题缩写	问题全称	single	multi	many	real	integer	label	binary	permutation	large	constrained	expensive	multimodal	sparse	dynamic	multitask	bilevel	robust
28	CEC2010_F12	CEC'2010 constrained optimization benchmark problem	√			√						√							
29	CEC2010_F13	CEC'2010 constrained optimization benchmark problem	√			√						√							
30	CEC2010_F14	CEC'2010 constrained optimization benchmark problem	√			√						√							
31	CEC2010_F15	CEC'2010 constrained optimization benchmark problem	√			√						√							
32	CEC2010_F16	CEC'2010 constrained optimization benchmark problem	√			√						√							
33	CEC2010_F17	CEC'2010 constrained optimization benchmark problem	√			√						√							
34	CEC2010_F18	CEC'2010 constrained optimization benchmark problem	√			√						√							
35	CEC2013_F1	Shifted elliptic function	√			√					√								
36	CEC2013_F2	Shifted Rastrigin's function	√			√					√								
37	CEC2013_F3	Shifted Ackley's function	√			√					√								
38	CEC2013_F4	7-nonseparable, 1-separable shifted and rotated elliptic function	√			√					√								
39	CEC2013_F5	7-nonseparable, 1-separable shifted and rotated Rastrigin's function	√			√					√								
40	CEC2013_F6	7-nonseparable, 1-separable shifted and rotated Ackley's function	√			√					√								
41	CEC2013_F7	7-nonseparable, 1-separable shifted and rotated Schwefel's function	√			√					√								
42	CEC2013_F8	20-nonseparable shifted and rotated elliptic function	√			√					√								
43	CEC2013_F9	20-nonseparable shifted and rotated Rastrigin's function	√			√					√								
44	CEC2013_F10	20-nonseparable shifted and rotated Rastrigin's function	√			√					√								
45	CEC2013_F11	20-nonseparable shifted and rotated Schwefel's function	√			√					√								
46	CEC2013_F12	Shifted Rosenbrock's function	√			√					√								
47	CEC2013_F13	Shifted Schwefel's function with conforming overlapping subcomponents	√			√					√								
48	CEC2013_F14	Shifted Schwefel's function with conflicting overlapping subcomponents	√			√					√								
49	CEC2013_F15	Shifted Schwefel's function	√			√					√								
50	CEC2017_F1	CEC'2017 constrained optimization benchmark problem	√			√						√							
51	CEC2017_F2	CEC'2017 constrained optimization benchmark problem	√			√						√							
52	CEC2017_F3	CEC'2017 constrained optimization benchmark problem	√			√						√							
53	CEC2017_F4	CEC'2017 constrained optimization benchmark problem	√			√						√							
54	CEC2017_F5	CEC'2017 constrained optimization benchmark problem	√			√						√							

	问题缩写	问题全称	single	multi	many	real	integer	label	binary	permutation	large	constrained	expensive	multimodal	sparse	dynamic	multitask	bilevel	robust
55	CEC2017_F6	CEC'2017 constrained optimization benchmark problem	√			√						√							
56	CEC2017_F7	CEC'2017 constrained optimization benchmark problem	√			√						√							
57	CEC2017_F8	CEC'2017 constrained optimization benchmark problem	√			√						√							
58	CEC2017_F9	CEC'2017 constrained optimization benchmark problem	√			√						√							
59	CEC2017_F10	CEC'2017 constrained optimization benchmark problem	√			√						√							
60	CEC2017_F11	CEC'2017 constrained optimization benchmark problem	√			√						√							
61	CEC2017_F12	CEC'2017 constrained optimization benchmark problem	√			√						√							
62	CEC2017_F13	CEC'2017 constrained optimization benchmark problem	√			√						√							
63	CEC2017_F14	CEC'2017 constrained optimization benchmark problem	√			√						√							
64	CEC2017_F15	CEC'2017 constrained optimization benchmark problem	√			√						√							
65	CEC2017_F16	CEC'2017 constrained optimization benchmark problem	√			√						√							
66	CEC2017_F17	CEC'2017 constrained optimization benchmark problem	√			√						√							
67	CEC2017_F18	CEC'2017 constrained optimization benchmark problem	√			√						√							
68	CEC2017_F19	CEC'2017 constrained optimization benchmark problem	√			√						√							
69	CEC2017_F20	CEC'2017 constrained optimization benchmark problem	√			√						√							
70	CEC2017_F21	CEC'2017 constrained optimization benchmark problem	√			√						√							
71	CEC2017_F22	CEC'2017 constrained optimization benchmark problem	√			√						√							
72	CEC2017_F23	CEC'2017 constrained optimization benchmark problem	√			√						√							
73	CEC2017_F24	CEC'2017 constrained optimization benchmark problem	√			√						√							
74	CEC2017_F25	CEC'2017 constrained optimization benchmark problem	√			√						√							
75	CEC2017_F26	CEC'2017 constrained optimization benchmark problem	√			√						√							
76	CEC2017_F27	CEC'2017 constrained optimization benchmark problem	√			√						√							
77	CEC2017_F28	CEC'2017 constrained optimization benchmark problem	√			√						√							
78	CEC2020_F1	Bent cigar function	√			√													
79	CEC2020_F2	Shifted and rotated Schwefel's function	√			√													

	问题缩写	问题全称	single	multi	many	real	integer	label	binary	permutation	large	constrained	expensive	multimodal	sparse	dynamic	multitask	bilevel	robust
80	CEC2020_F3	Shifted and rotated Lunacek bi-Rastrigin function	√			√													
81	CEC2020_F4	Expanded Rosenbrock's plus Griewangk's function	√			√													
82	CEC2020_F5	Hybrid function 1	√			√													
83	CEC2020_F6	Hybrid function 2	√			√													
84	CEC2020_F7	Hybrid function 3	√			√													
85	CEC2020_F8	Composition function 1	√			√													
86	CEC2020_F9	Composition function 2	√			√													
87	CEC2020_F10	Composition function 3	√			√													
88	CF1	Constrained benchmark MOP		√		√					√	√							
89	CF2	Constrained benchmark MOP		√		√					√	√							
90	CF3	Constrained benchmark MOP		√		√					√	√							
91	CF4	Constrained benchmark MOP		√		√					√	√							
92	CF5	Constrained benchmark MOP		√		√					√	√							
93	CF6	Constrained benchmark MOP		√		√					√	√							
94	CF7	Constrained benchmark MOP		√		√					√	√							
95	CF8	Constrained benchmark MOP		√		√					√	√							
96	CF9	Constrained benchmark MOP		√		√					√	√							
97	CF10	Constrained benchmark MOP		√		√					√	√							
98	CI_HS	Multitasking problem (Griewank function + Rastrigin function)	√			√					√						√		
99	CI_LS	Multitasking problem (Ackley function + Schwefel function)	√			√					√						√		
100	CI_MS	Multitasking problem (Ackley function + Rastrigin function)	√			√					√						√		
101	Community Detection	The community detection problem with label based encoding	√					√			√		√						
102	DAS-CMOP1	Difficulty-adjustable and scalable constrained benchmark MOP		√		√					√	√							
103	DAS-CMOP2	Difficulty-adjustable and scalable constrained benchmark MOP		√		√					√	√							
104	DAS-CMOP3	Difficulty-adjustable and scalable constrained benchmark MOP		√		√					√	√							
105	DAS-CMOP4	Difficulty-adjustable and scalable constrained benchmark MOP		√		√					√	√							
106	DAS-CMOP5	Difficulty-adjustable and scalable constrained benchmark MOP		√		√					√	√							
107	DAS-CMOP6	Difficulty-adjustable and scalable constrained benchmark MOP		√		√					√	√							
108	DAS-CMOP7	Difficulty-adjustable and scalable constrained benchmark MOP		√		√					√	√							
109	DAS-CMOP8	Difficulty-adjustable and scalable constrained benchmark MOP		√		√					√	√							

	问题缩写	问题全称	single	multi	many	real	integer	label	binary	permutation	large	constrained	expensive	multimodal	sparse	dynamic	multitask	bilevel	robust
110	DAS-CMOP9	Difficulty-adjustable and scalable constrained benchmark MOP		√		√					√	√							
111	DOC1	Benchmark MOP with constraints in decision and objective spaces		√		√						√							
112	DOC2	Benchmark MOP with constraints in decision and objective spaces		√		√						√							
113	DOC3	Benchmark MOP with constraints in decision and objective spaces		√		√						√							
114	DOC4	Benchmark MOP with constraints in decision and objective spaces		√		√						√							
115	DOC5	Benchmark MOP with constraints in decision and objective spaces		√		√						√							
116	DOC6	Benchmark MOP with constraints in decision and objective spaces		√		√						√							
117	DOC7	Benchmark MOP with constraints in decision and objective spaces		√		√						√							
118	DOC8	Benchmark MOP with constraints in decision and objective spaces		√		√						√							
119	DOC9	Benchmark MOP with constraints in decision and objective spaces		√		√						√							
120	DTLZ1	Benchmark MOP proposed by Deb, Thiele, Laumanns, and Zitzler		√	√	√					√		√						
121	DTLZ2	Benchmark MOP proposed by Deb, Thiele, Laumanns, and Zitzler		√	√	√					√		√						
122	DTLZ3	Benchmark MOP proposed by Deb, Thiele, Laumanns, and Zitzler		√	√	√					√		√						
123	DTLZ4	Benchmark MOP proposed by Deb, Thiele, Laumanns, and Zitzler		√	√	√					√		√						
124	DTLZ5	Benchmark MOP proposed by Deb, Thiele, Laumanns, and Zitzler		√	√	√					√		√						
125	DTLZ6	Benchmark MOP proposed by Deb, Thiele, Laumanns, and Zitzler		√	√	√					√		√						
126	DTLZ7	Benchmark MOP proposed by Deb, Thiele, Laumanns, and Zitzler		√	√	√					√		√						
127	DTLZ8	Benchmark MOP proposed by Deb, Thiele, Laumanns, and Zitzler		√	√	√					√	√	√						
128	DTLZ9	Benchmark MOP proposed by Deb, Thiele, Laumanns, and Zitzler		√	√	√					√	√	√						
129	CDTLZ2	Convex DTLZ2		√	√	√					√		√						
130	IDTLZ1	Inverted DTLZ1		√	√	√					√		√						
131	IDTLZ2	Inverted DTLZ2		√	√	√					√		√						
132	SDTLZ1	Scaled DTLZ1		√	√	√					√		√						
133	SDTLZ2	Scaled DTLZ2		√	√	√					√		√						
134	C1-DTLZ1	Constrained DTLZ1		√	√	√					√	√	√						
135	C1-DTLZ3	Constrained DTLZ3		√	√	√					√	√	√						
136	C2-DTLZ2	Constrained DTLZ2		√	√	√					√	√	√						

	问题缩写	问题全称	single	multi	many	real	integer	label	binary	permutation	large	constrained	expensive	multimodal	sparse	dynamic	multitask	bilevel	robust
137	C3-DTLZ4	Constrained DTLZ4		√	√	√					√	√	√						
138	DC1-DTLZ1	DTLZ1 with constrains in decision space		√	√	√					√	√	√						
139	DC1-DTLZ3	DTLZ3 with constrains in decision space		√	√	√					√	√	√						
140	DC2-DTLZ1	DTLZ1 with constrains in decision space		√	√	√					√	√	√						
141	DC2-DTLZ3	DTLZ3 with constrains in decision space		√	√	√					√	√	√						
142	DC3-DTLZ1	DTLZ1 with constrains in decision space		√	√	√					√	√	√						
143	DC3-DTLZ3	DTLZ3 with constrains in decision space		√	√	√					√	√	√						
144	FCP1	Benchmark constrained MOP proposed by Yuan		√		√						√							
145	FCP2	Benchmark constrained MOP proposed by Yuan		√		√						√							
146	FCP3	Benchmark constrained MOP proposed by Yuan		√		√						√							
147	FCP4	Benchmark constrained MOP proposed by Yuan		√		√						√							
148	FCP5	Benchmark constrained MOP proposed by Yuan		√		√						√							
149	FDA1	Benchmark dynamic MOP proposed by Farina, Deb, and Amato		√		√					√					√			
150	FDA2	Benchmark dynamic MOP proposed by Farina, Deb, and Amato		√		√					√					√			
151	FDA3	Benchmark dynamic MOP proposed by Farina, Deb, and Amato		√		√					√					√			
152	FDA4	Benchmark dynamic MOP proposed by Farina, Deb, and Amato		√		√					√					√			
153	FDA5	Benchmark dynamic MOP proposed by Farina, Deb, and Amato		√		√					√					√			
154	IMMOEA_F1	Benchmark MOP for testing IM-MOEA		√		√					√								
155	IMMOEA_F2	Benchmark MOP for testing IM-MOEA		√		√					√								
156	IMMOEA_F3	Benchmark MOP for testing IM-MOEA		√		√					√								
157	IMMOEA_F4	Benchmark MOP for testing IM-MOEA		√		√					√								
158	IMMOEA_F5	Benchmark MOP for testing IM-MOEA		√		√					√								
159	IMMOEA_F6	Benchmark MOP for testing IM-MOEA		√		√					√								
160	IMMOEA_F7	Benchmark MOP for testing IM-MOEA		√		√					√								
161	IMMOEA_F8	Benchmark MOP for testing IM-MOEA		√		√					√								
162	IMMOEA_F9	Benchmark MOP for testing IM-MOEA		√		√					√								
163	IMMOEA_F10	Benchmark MOP for testing IM-MOEA		√		√					√								
164	IMOP1	Benchmark MOP with irregular Pareto front		√		√							√						
165	IMOP2	Benchmark MOP with irregular Pareto front		√		√							√						
166	IMOP3	Benchmark MOP with irregular Pareto front		√		√							√						
167	IMOP4	Benchmark MOP with irregular Pareto front		√		√							√						
168	IMOP5	Benchmark MOP with irregular Pareto front		√		√							√						
169	IMOP6	Benchmark MOP with irregular Pareto front		√		√							√						
170	IMOP7	Benchmark MOP with irregular Pareto front		√		√							√						
171	IMOP8	Benchmark MOP with irregular Pareto front		√		√							√						

	问题缩写	问题全称	single	multi	many	real	integer	label	binary	permutation	large	constrained	expensive	multimodal	sparse	dynamic	multitask	bilevel	robust
172	Instance1	Multitasking multi-objective problem (ZDT4-R + ZDT4-G)		√		√					√						√		
173	Instance2	Multitasking multi-objective problem (ZDT4-RC + ZDT4-A)		√		√					√	√					√		
174	KP	The knapsack problem	√						√		√	√							
175	LIR-CMOP1	Constrained benchmark MOP with large infeasible regions		√		√					√	√							
176	LIR-CMOP2	Constrained benchmark MOP with large infeasible regions		√		√					√	√							
177	LIR-CMOP3	Constrained benchmark MOP with large infeasible regions		√		√					√	√							
178	LIR-CMOP4	Constrained benchmark MOP with large infeasible regions		√		√					√	√							
179	LIR-CMOP5	Constrained benchmark MOP with large infeasible regions		√		√					√	√							
180	LIR-CMOP6	Constrained benchmark MOP with large infeasible regions		√		√					√	√							
181	LIR-CMOP7	Constrained benchmark MOP with large infeasible regions		√		√					√	√							
182	LIR-CMOP8	Constrained benchmark MOP with large infeasible regions		√		√					√	√							
183	LIR-CMOP9	Constrained benchmark MOP with large infeasible regions		√		√					√	√							
184	LIR-CMOP10	Constrained benchmark MOP with large infeasible regions		√		√					√	√							
185	LIR-CMOP11	Constrained benchmark MOP with large infeasible regions		√		√					√	√							
186	LIR-CMOP12	Constrained benchmark MOP with large infeasible regions		√		√					√	√							
187	LIR-CMOP13	Constrained benchmark MOP with large infeasible regions		√		√					√	√							
188	LIR-CMOP14	Constrained benchmark MOP with large infeasible regions		√		√					√	√							
189	LSMOP1	Large-scale benchmark MOP		√	√	√					√								
190	LSMOP2	Large-scale benchmark MOP		√	√	√					√								
191	LSMOP3	Large-scale benchmark MOP		√	√	√					√								
192	LSMOP4	Large-scale benchmark MOP		√	√	√					√								
193	LSMOP5	Large-scale benchmark MOP		√	√	√					√								
194	LSMOP6	Large-scale benchmark MOP		√	√	√					√								
195	LSMOP7	Large-scale benchmark MOP		√	√	√					√								
196	LSMOP8	Large-scale benchmark MOP		√	√	√					√								
197	LSMOP9	Large-scale benchmark MOP		√	√	√					√								
198	MaF1	Inverted DTLZ1		√	√	√					√								
199	MaF2	DTLZ2BZ		√	√	√					√								
200	MaF3	Convex DTLZ3		√	√	√					√								

	问题缩写	问题全称	single	multi	many	real	integer	label	binary	permutation	large	constrained	expensive	multimodal	sparse	dynamic	multitask	bilevel	robust
201	MaF4	Inverted and scaled DTLZ3		√	√	√					√								
202	MaF5	Scaled DTLZ4		√	√	√					√								
203	MaF6	DTLZ5IM		√	√	√					√								
204	MaF7	DTLZ7		√	√	√					√								
205	MaF8	MP-DMP		√	√	√													
206	MaF9	ML-DMP		√	√	√													
207	MaF10	WFG1		√	√	√					√								
208	MaF11	WFG2		√	√	√					√								
209	MaF12	WFG9		√	√	√					√								
210	MaF13	P7		√	√	√					√								
211	MaF14	LSMOP3		√	√	√					√								
212	MaF15	Inverted LSMOP8		√	√	√					√								
213	MaOPP_binary	Many-objective pathfinding problem based on binary encoding			√				√		√		√						
214	MaOPP_real	Many-objective pathfinding problem based on real encoding			√	√					√		√						
215	MLDMP	The multi-line distance minimization problem		√	√	√													
216	MMF1	Multi-modal multi-objective test function		√		√								√					
217	MMF2	Multi-modal multi-objective test function		√		√								√					
218	MMF3	Multi-modal multi-objective test function		√		√								√					
219	MMF4	Multi-modal multi-objective test function		√		√								√					
220	MMF5	Multi-modal multi-objective test function		√		√								√					
221	MMF6	Multi-modal multi-objective test function		√		√								√					
222	MMF7	Multi-modal multi-objective test function		√		√								√					
223	MMF8	Multi-modal multi-objective test function		√		√								√					
224	MMMOP1	Multi-modal multi-objective optimization problem		√	√	√								√					
225	MMMOP2	Multi-modal multi-objective optimization problem		√	√	√								√					
226	MMMOP3	Multi-modal multi-objective optimization problem		√	√	√								√					
227	MMMOP4	Multi-modal multi-objective optimization problem		√	√	√								√					
228	MMMOP5	Multi-modal multi-objective optimization problem		√	√	√								√					
229	MMMOP6	Multi-modal multi-objective optimization problem		√	√	√								√					
230	MOEADDE_F1	Benchmark MOP for testing MOEA/D-DE		√		√					√								
231	MOEADDE_F2	Benchmark MOP for testing MOEA/D-DE		√		√					√								
232	MOEADDE_F3	Benchmark MOP for testing MOEA/D-DE		√		√					√								
233	MOEADDE_F4	Benchmark MOP for testing MOEA/D-DE		√		√					√								
234	MOEADDE_F5	Benchmark MOP for testing MOEA/D-DE		√		√					√								
235	MOEADDE_F6	Benchmark MOP for testing MOEA/D-DE		√		√					√								
236	MOEADDE_F7	Benchmark MOP for testing MOEA/D-DE		√		√					√								

	问题缩写	问题全称	single	multi	many	real	integer	label	binary	permutation	large	constrained	expensive	multimodal	sparse	dynamic	multitask	bilevel	robust
237	MOEADDE_F8	Benchmark MOP for testing MOEA/D-DE		√		√					√								
238	MOEADDE_F9	Benchmark MOP for testing MOEA/D-DE		√		√					√								
239	MOEADM2M_F1	Benchmark MOP for testing MOEA/D-M2M		√		√					√								
240	MOEADM2M_F2	Benchmark MOP for testing MOEA/D-M2M		√		√					√								
241	MOEADM2M_F3	Benchmark MOP for testing MOEA/D-M2M		√		√					√								
242	MOEADM2M_F4	Benchmark MOP for testing MOEA/D-M2M		√		√					√								
243	MOEADM2M_F5	Benchmark MOP for testing MOEA/D-M2M		√		√					√								
244	MOEADM2M_F6	Benchmark MOP for testing MOEA/D-M2M		√		√					√								
245	MOEADM2M_F7	Benchmark MOP for testing MOEA/D-M2M		√		√					√								
246	MOKP	The multi-objective knapsack problem		√	√				√		√								
247	MONRP	The multi-objective next release problem		√					√		√								
248	MOTSP	The multi-objective traveling salesman problem		√	√					√	√								
249	MPDMP	The multi-point distance minimization problem		√	√	√													
250	mQAP	The multi-objective quadratic assignment problem		√	√					√	√								
251	MW1	Constrained benchmark MOP proposed by Ma and Wang		√		√					√	√							
252	MW2	Constrained benchmark MOP proposed by Ma and Wang		√		√					√	√							
253	MW3	Constrained benchmark MOP proposed by Ma and Wang		√		√					√	√							
254	MW4	Constrained benchmark MOP proposed by Ma and Wang		√	√	√					√	√							
255	MW5	Constrained benchmark MOP proposed by Ma and Wang		√		√					√	√							
256	MW6	Constrained benchmark MOP proposed by Ma and Wang		√		√					√	√							
257	MW7	Constrained benchmark MOP proposed by Ma and Wang		√		√					√	√							
258	MW8	Constrained benchmark MOP proposed by Ma and Wang		√	√	√					√	√							
259	MW9	Constrained benchmark MOP proposed by Ma and Wang		√		√					√	√							
260	MW10	Constrained benchmark MOP proposed by Ma and Wang		√		√					√	√							
261	MW11	Constrained benchmark MOP proposed by Ma and Wang		√		√					√	√							
262	MW12	Constrained benchmark MOP proposed by Ma and Wang		√		√					√	√							
263	MW13	Constrained benchmark MOP proposed by Ma and Wang		√		√					√	√							
264	MW14	Constrained benchmark MOP proposed by Ma and Wang		√	√	√					√	√							
265	NI_HS	Multitasking problem (Rosenbrock function + Rastrigin function)	√			√					√						√		

	问题缩写	问题全称	single	multi	many	real	integer	label	binary	permutation	large	constrained	expensive	multimodal	sparse	dynamic	multitask	bilevel	robust
266	NI_MS	Multitasking problem (Griewank function + Weierstrass function)	√			√					√						√		
267	RMMEDA_F1	Benchmark MOP for testing RM-MEDA		√		√					√								
268	RMMEDA_F2	Benchmark MOP for testing RM-MEDA		√		√					√								
269	RMMEDA_F3	Benchmark MOP for testing RM-MEDA		√		√					√								
270	RMMEDA_F4	Benchmark MOP for testing RM-MEDA		√		√					√								
271	RMMEDA_F5	Benchmark MOP for testing RM-MEDA		√		√					√								
272	RMMEDA_F6	Benchmark MOP for testing RM-MEDA		√		√					√								
273	RMMEDA_F7	Benchmark MOP for testing RM-MEDA		√		√					√								
274	RMMEDA_F8	Benchmark MOP for testing RM-MEDA		√		√					√								
275	RMMEDA_F9	Benchmark MOP for testing RM-MEDA		√		√					√								
276	RMMEDA_F10	Benchmark MOP for testing RM-MEDA		√		√					√								
277	RWMOP1	Pressure vessal problem		√		√						√							
278	RWMOP2	Vibrating platform		√		√						√							
279	RWMOP3	Two bar truss design problem		√		√						√							
280	RWMOP4	Weldan beam design problem		√		√						√							
281	RWMOP5	Disc brake design problem		√		√						√							
282	RWMOP6	Speed reducer design problem		√		√						√							
283	RWMOP7	Gear train design problem		√		√						√							
284	RWMOP8	Car side impact design problem		√		√						√							
285	RWMOP9	Four bar plane truss		√		√						√							
286	RWMOP10	Two bar plane truss		√		√						√							
287	RWMOP11	Water resource management problem		√		√						√							
288	RWMOP12	Simply supported I-beam design		√		√						√							
289	RWMOP13	Gear box design		√		√						√							
290	RWMOP14	Multiple-disk clutch brake design problem		√		√						√							
291	RWMOP15	Spring design problem		√		√						√							
292	RWMOP16	Cantilever beam design problem		√		√						√							
293	RWMOP17	Bulk carriers design problem		√		√						√							
294	RWMOP18	Front rail design problem		√		√						√							
295	RWMOP19	Multi-product batch plant		√		√						√							
296	RWMOP20	Hydro-static thrust bearing design problem		√		√						√							
297	RWMOP21	Crash energy management for high-speed train		√		√						√							
298	RWMOP22	Haverly's pooling problem		√		√						√							
299	RWMOP23	Reactor network design		√		√						√							
300	RWMOP24	Heat exchanger network design		√		√						√							
301	RWMOP25	Process synthesis problem		√		√						√							
302	RWMOP26	Process sythesis and design problem		√		√						√							

	问题缩写	问题全称	single	multi	many	real	integer	label	binary	permutation	large	constrained	expensive	multimodal	sparse	dynamic	multitask	bilevel	robust
303	RWMOP27	Process flow sheeting problem		√		√						√							
304	RWMOP28	Two reactor problem		√		√						√							
305	RWMOP29	Process synthesis problem		√		√						√							
306	RWMOP30	Synchronous pptimal pulse-width modulation of 3-level inverters		√		√						√							
307	RWMOP31	Synchronous pptimal pulse-width modulation of 5-level inverters		√		√						√							
308	RWMOP32	Synchronous pptimal pulse-width modulation of 7-level inverters		√		√						√							
309	RWMOP33	Synchronous pptimal pulse-width modulation of 9-level inverters		√		√						√							
310	RWMOP34	Synchronous pptimal pulse-width modulation of 11-level inverters		√		√						√							
311	RWMOP35	Synchronous pptimal pulse-width modulation of 13-level inverters		√		√						√							
312	RWMOP36	Optimal sizing of single phase distributed generation with reactive power support for phase balancing at main transformer/grid and active power loss		√		√						√							
313	RWMOP37	Optimal Sizing of Single Phase Distributed Generation with reactive power support for Phase Balancing at Main Transformer/Grid and reactive Power loss		√		√						√							
314	RWMOP38	Optimal sizing of single phase distributed generation with reactive power support for active and reactive power loss		√		√						√							
315	RWMOP39	Optimal sizing of single phase distributed generation with reactive power support for phase balancing at main transformer/grid and active and reactive power loss		√		√						√							
316	RWMOP40	Optimal power flow for minimizing active and reactive power loss		√		√						√							
317	RWMOP41	Optimal power flow for minimizing voltage deviation, active and reactive power loss		√		√						√							
318	RWMOP42	Optimal power flow for minimizing voltage deviation, and active power loss		√		√						√							
319	RWMOP43	Optimal power flow for minimizing fuel cost, and active power loss		√		√						√							
320	RWMOP44	Optimal power flow for minimizing fuel cost, active and reactive power loss		√		√						√							
321	RWMOP45	Optimal power flow for minimizing fuel cost, voltage deviation, and active power loss		√		√						√							
322	RWMOP46	Optimal power flow for minimizing fuel cost, voltage deviation, active and reactive power loss		√		√						√							
323	RWMOP47	Optimal droop setting for minimizing active and reactive power loss		√		√						√							
324	RWMOP48	Optimal droop setting for minimizing voltage deviation and active power loss		√		√						√							
325	RWMOP49	Optimal droop setting for minimizing voltage deviation, active, and reactive power loss		√		√						√							

	问题缩写	问题全称	single	multi	many	real	integer	label	binary	permutation	large	constrained	expensive	multimodal	sparse	dynamic	multitask	bilevel	robust
326	RWMOP50	Power distribution system planning		√		√						√							
327	SMD1	Bilevel optimization problems proposed by Sinha, Malo, and Deb		√		√												√	
328	SMD2	Bilevel optimization problems proposed by Sinha, Malo, and Deb		√		√												√	
329	SMD3	Bilevel optimization problems proposed by Sinha, Malo, and Deb		√		√												√	
330	SMD4	Bilevel optimization problems proposed by Sinha, Malo, and Deb		√		√												√	
331	SMD5	Bilevel optimization problems proposed by Sinha, Malo, and Deb		√		√												√	
332	SMD6	Bilevel optimization problems proposed by Sinha, Malo, and Deb		√		√												√	
333	SMD7	Bilevel optimization problems proposed by Sinha, Malo, and Deb		√		√												√	
334	SMD8	Bilevel optimization problems proposed by Sinha, Malo, and Deb		√		√												√	
335	SMD9	Bilevel optimization problems proposed by Sinha, Malo, and Deb		√		√						√						√	
336	SMD10	Bilevel optimization problems proposed by Sinha, Malo, and Deb		√		√						√						√	
337	SMD11	Bilevel optimization problems proposed by Sinha, Malo, and Deb		√		√						√						√	
338	SMD12	Bilevel optimization problems proposed by Sinha, Malo, and Deb		√		√						√						√	
339	Sparse_CD	The community detection problem		√					√		√		√		√				
340	Sparse_CN	The critical node detection problem		√					√		√		√		√				
341	Sparse_FS	The feature selection problem		√					√		√		√		√				
342	Sparse_IS	The instance selection problem		√					√		√		√		√				
343	Sparse_KP	The sparse multi-objective knapsack problem		√	√				√		√								
344	Sparse_NN	The neural network training problem		√		√					√		√		√				
345	Sparse_PM	The pattern mining problem		√					√		√		√		√				
346	Sparse_PO	The portfolio optimization problem		√		√					√		√		√				
347	Sparse_SR	The sparse signal reconstruction problem		√		√					√		√		√				
348	SMMOP1	Sparse multi-modal multi-objective optimization problem		√	√	√					√			√	√				
349	SMMOP2	Sparse multi-modal multi-objective optimization problem		√	√	√					√			√	√				
350	SMMOP3	Sparse multi-modal multi-objective optimization problem		√	√	√					√			√	√				
351	SMMOP4	Sparse multi-modal multi-objective optimization problem		√	√	√					√			√	√				
352	SMMOP5	Sparse multi-modal multi-objective optimization problem		√	√	√					√			√	√				
353	SMMOP6	Sparse multi-modal multi-objective optimization problem		√	√	√					√			√	√				

	问题缩写	问题全称	single	multi	many	real	integer	label	binary	permutation	large	constrained	expensive	multimodal	sparse	dynamic	multitask	bilevel	robust
354	SMMOP7	Sparse multi-modal multi-objective optimization problem		√	√	√					√			√	√				
355	SMMOP8	Sparse multi-modal multi-objective optimization problem		√	√	√					√			√	√				
356	SMOP1	Benchmark MOP with sparse Pareto optimal solutions		√	√	√					√		√		√				
357	SMOP2	Benchmark MOP with sparse Pareto optimal solutions		√	√	√					√		√		√				
358	SMOP3	Benchmark MOP with sparse Pareto optimal solutions		√	√	√					√		√		√				
359	SMOP4	Benchmark MOP with sparse Pareto optimal solutions		√	√	√					√		√		√				
360	SMOP5	Benchmark MOP with sparse Pareto optimal solutions		√	√	√					√		√		√				
361	SMOP6	Benchmark MOP with sparse Pareto optimal solutions		√	√	√					√		√		√				
362	SMOP7	Benchmark MOP with sparse Pareto optimal solutions		√	√	√					√		√		√				
363	SMOP8	Benchmark MOP with sparse Pareto optimal solutions		√	√	√					√		√		√				
364	SOP_F1	Sphere function	√			√							√						
365	SOP_F2	Schwefel's function 2.22	√			√							√						
366	SOP_F3	Schwefel's function 1.2	√			√							√						
367	SOP_F4	Schwefel's function 2.21	√			√							√						
368	SOP_F5	Generalized Rosenbrock's function	√			√							√						
369	SOP_F6	Step function	√			√							√						
370	SOP_F7	Quartic function with noise	√			√							√						
371	SOP_F8	Generalized Schwefel's function 2.26	√			√							√						
372	SOP_F9	Generalized Rastrigin's function	√			√							√						
373	SOP_F10	Ackley's function	√			√							√						
374	SOP_F11	Generalized Griewank's function	√			√							√						
375	SOP_F12	Generalized penalized function	√			√							√						
376	SOP_F13	Generalized penalized function	√			√							√						
377	SOP_F14	Shekel's foxholes function	√			√							√						
378	SOP_F15	Kowalik's function	√			√							√						
379	SOP_F16	Six-hump camel-back function	√			√							√						
380	SOP_F17	Branin function	√			√							√						
381	SOP_F18	Goldstein-price function	√			√							√						
382	SOP_F19	Hartman's family	√			√							√						
383	SOP_F20	Hartman's family	√			√							√						
384	SOP_F21	Shekel's family	√			√							√						
385	SOP_F22	Shekel's family	√			√							√						

	问题缩写	问题全称	single	multi	many	real	integer	label	binary	permutation	large	constrained	expensive	multimodal	sparse	dynamic	multitask	bilevel	robust
386	SOP_F23	Shekel's family	√			√							√						
387	TP1	Test problem for robust multi-objective optimization		√		√					√								√
388	TP2	Test problem for robust multi-objective optimization		√		√					√								√
389	TP3	Test problem for robust multi-objective optimization		√		√					√								√
390	TP4	Test problem for robust multi-objective optimization		√		√					√								√
391	TP5	Test problem for robust multi-objective optimization		√		√					√								√
392	TP6	Test problem for robust multi-objective optimization		√		√					√								√
393	TP7	Test problem for robust multi-objective optimization		√		√					√								√
394	TP8	Test problem for robust multi-objective optimization		√		√					√								√
395	TP9	Test problem for robust multi-objective optimization		√		√					√								√
396	TP10	Test problem for robust multi-objective optimization		√		√					√	√							√
397	TREE1	The time-varying ratio error estimation problem		√		√					√	√	√						
398	TREE2	The time-varying ratio error estimation problem		√		√					√	√	√						
399	TREE3	The time-varying ratio error estimation problem		√		√					√	√	√						
400	TREE4	The time-varying ratio error estimation problem		√		√					√	√	√						
401	TREE5	The time-varying ratio error estimation problem		√		√					√	√	√						
402	TREE6	The time-varying ratio error estimation problem		√		√					√	√	√						
403	TSP	The traveling salesman problem	√							√	√								
404	UF1	Unconstrained benchmark MOP		√		√					√								
405	UF2	Unconstrained benchmark MOP		√		√					√								
406	UF3	Unconstrained benchmark MOP		√		√					√								
407	UF4	Unconstrained benchmark MOP		√		√					√								
408	UF5	Unconstrained benchmark MOP		√		√					√								
409	UF6	Unconstrained benchmark MOP		√		√					√								
410	UF7	Unconstrained benchmark MOP		√		√					√								
411	UF8	Unconstrained benchmark MOP		√		√					√								
412	UF9	Unconstrained benchmark MOP		√		√					√								
413	UF10	Unconstrained benchmark MOP		√		√					√								
414	VNT1	Benchmark MOP proposed by Viennet		√		√													
415	VNT2	Benchmark MOP proposed by Viennet		√		√													
416	VNT3	Benchmark MOP proposed by Viennet		√		√													
417	VNT4	Benchmark MOP proposed by Viennet		√		√						√							
418	WFG1	Benchmark MOP proposed by Walking Fish Group		√	√	√					√		√						
419	WFG2	Benchmark MOP proposed by Walking Fish Group		√	√	√					√		√						
420	WFG3	Benchmark MOP proposed by Walking Fish Group		√	√	√					√		√						
421	WFG4	Benchmark MOP proposed by Walking Fish Group		√	√	√					√		√						
422	WFG5	Benchmark MOP proposed by Walking Fish Group		√	√	√					√		√						

	问题缩写	问题全称	single	multi	many	real	integer	label	binary	permutation	large	constrained	expensive	multimodal	sparse	dynamic	multitask	bilevel	robust
423	WFG6	Benchmark MOP proposed by Walking Fish Group		√	√	√					√		√						
424	WFG7	Benchmark MOP proposed by Walking Fish Group		√	√	√					√		√						
425	WFG8	Benchmark MOP proposed by Walking Fish Group		√	√	√					√		√						
426	WFG9	Benchmark MOP proposed by Walking Fish Group		√	√	√					√		√						
427	ZDT1	Benchmark MOP proposed by Zitzler, Deb, and Thiele		√		√					√		√						
428	ZDT2	Benchmark MOP proposed by Zitzler, Deb, and Thiele		√		√					√		√						
429	ZDT3	Benchmark MOP proposed by Zitzler, Deb, and Thiele		√		√					√		√						
430	ZDT4	Benchmark MOP proposed by Zitzler, Deb, and Thiele		√		√					√		√						
431	ZDT5	Benchmark MOP proposed by Zitzler, Deb, and Thiele		√					√		√		√						
432	ZDT6	Benchmark MOP proposed by Zitzler, Deb, and Thiele		√		√					√		√						