

多目标优化算法在多分类中的应用研究

尚荣华, 胡朝旭, 焦李成, 白 靖

(西安电子科技大学智能感知与图像理解教育部重点实验室, 陕西西安 710071)

摘 要: Cai 等人用多目标粒子群算法(MOPSO)优化多目标聚类学习和分类学习框架(MSCC)的多目标问题时, 种群只能得到少量的非支配解, 不利于种群优化. 而在此情况下, NSGA-II 由于采用了 Pareto 排序的方法, 种群中会保留大量优秀的支配解, 有利于种群优化, 所以本文引进了 NSGA-II 优化 MSCC 框架的多目标问题. 通过对数据集的测试, 验证了在 NSGA-II 的优化下, 对于大多数测试问题, MSCC 框架设计的分类器的最大分类正确率高于 MOPSO 优化 MSCC 框架的结果. 进而对实验结果做了进一步分析, 发现了最大正确率不随多目标优化算法的优化过程而提高的问题.

关键词: 多分类; 多目标优化; 聚类; MOPSO; NSGA-II

中图分类号: TP18 **文献标识码:** A **文章编号:** 0372-2112 (2012) 11-2264-06

电子学报 URL: <http://www.ejournal.org.cn> **DOI:** 10.3969/j.issn.0372-2112.2012.11.019

Research of Multi-Objective Optimization Algorithms' Application in Multi-Class Classification

SHANG Rong-hua, HU Chao-xu, JIAO Li-cheng, BAI Jing

(Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education of China, Xidian University, Xi'an, Shaanxi 710071, China)

Abstract: When Multi-objective Particle Swarm Optimization (MOPSO) optimizes the multi-objective problems of the multi-objective simultaneous learning framework (MSCC), there are only a few nondominated solutions in MOPSO population. In this case, NSGA-II can keep a lot of good dominated solutions in the population, which will help the population optimize, so this paper brought in NSGA-II as the optimization algorithm. The results of experiments show that, under the optimization of NSGA-II, MSCC framework can get better multi-class classifiers. However, dominated solutions can get better classifiers than nondominated solutions. By observing the changing curves of the maximum classification accuracy rate following with the optimization of populations, this paper found that, when dealing with most of the data sets, the maximum accuracy is not improved following the optimization of populations.

Key words: multi-class; multi-objective optimization; cluster; MOPSO; NSGA-II

1 引言

多分类是数据挖掘领域最重要的分支之一, 多年来, 众多学者已经提出了许多经典的多分类算法, 如 RBFNN (Radial Basis Function Neural Networks)^[1~3], SVM (Support Vector Machines)^[4~10], C4.5^[11~13]等. RBFNN, SVM 和 C4.5 算法的示意图如图 1 所示, 首先在给定的训练集上进行训练, 得到分类器后在测试数据集上进行测试, 通过所得的分类正确率来判断分类算法的好坏.

2010 年, Cai 等人^[14]首次在训练集训练分类器的过

程中提出了一个同时聚类和分类的框架(MSCC), MSCC 从多目标的角度来刻画聚类和分类问题, 为了实现多目标学习, Cai 等人借鉴 SCC 中设计的结合结构信息的分类器, 使得 MSCC 中的多个目标函数都仅依赖于聚类中心. 并在此基础上采用了两个最小化目标函数: 聚类目标函数和分类目标函数^[14]. 通过聚类中心计算两个目标函数的值, 采用 MOPSO^[15]优化这两个目标函数, 最终得到多个分类器, 通过对测试数据集进行测试, 选取得到最大分类正确率的分类器, 取得了较好的效果. 然而, 文献[14]中指出, MOPSO 优化 MSCC 框架的多目标问题

收稿日期: 2011-12-23; 修回日期: 2012-04-10

基金项目: 国家自然科学基金(No. 61001202, No. 61072139, No. 61003199); 中国博士后科学基金(No. 201104658, No. 20090451369, No. 20090461283); 陕西省自然科学基金基础研究计划(No. 2010JQ8023, No. 2011JQ8010); 国家教育部博士点基金(No. 20100203120008, No. 20090203120016, No. 200807010003); 高等学校学科创新引智计划(No. B07048); 教育部“长江学者和创新团队发展计划”(No. IRT1170)

时,仅能得到少量的非支配解.而在这种情况下,NSGA-II^[16]由于采用了 Pareto 排序方法,种群保存大量优秀的支配解,这样有利于种群优化,所以本文引进了 NSGA-II 算法优化 MSCC 框架中的多目标问题,并对多目标优化算法对 MSCC 框架的作用做了进一步的研究.实验结果表明,新方法得到了较好的效果.

2 多目标优化及相关概念

为了方便问题的阐述,首先给出一个标准的无约束多目标问题^[16]:

$$\begin{aligned} \text{Min } \mathbf{F}(\mathbf{x}) &= (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_k(\mathbf{x})) \\ \text{s.t. } \mathbf{x} &= (x_1, x_2, \dots, x_m) \in X \\ \mathbf{u} &= (u_1, u_2, \dots, u_m), \mathbf{l} = (l_1, l_2, \dots, l_m) \end{aligned} \quad (1)$$

其中, $\mathbf{x} = (x_1, x_2, \dots, x_m)$ 为 m 维决策向量, x_j 是第 j 维决策变量, X 为决策变量空间, $\mathbf{u}$ 和 $\mathbf{l}$ 为决策变量的上下界. $\mathbf{F}(\mathbf{x})$ 为目标函数向量, $f_i(\mathbf{x})$ 为第 i 维目标函数值.多目标优化算法的示意图如图 2 所示.

首先对种群初始化,之后对种群进行优化操作,更新种群,不断迭代,直到结束条件满足后,输出最优结果.其中,NSGA-II 为最具代表性的多目

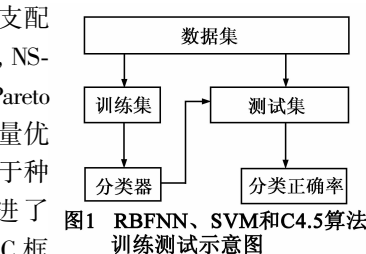


图1 RBFNN、SVM和C4.5算法训练测试示意图

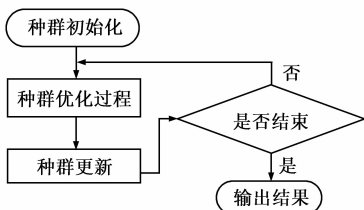


图2 多目标优化算法示意图

标优化算法之一.通过非支配排序算法将种群中的个体进行非支配排序,计算拥挤度值,然后选取非支配等级小且拥挤度值大的个体进入下一代进行进化.以下介绍两个概念:Pareto 占优和非支配解.

2.1 Pareto 占优

如果 $\mathbf{x}, \mathbf{x}^* \in X_f$, $\mathbf{x}^*$ 是 Pareto 占优的,当且仅当式(2)成立,记作 $\mathbf{x}^* > \mathbf{x}$.

$$\begin{aligned} (\forall i \in \{1, \dots, k\}: f_i(\mathbf{x}^*) \leq f_i(\mathbf{x})) \wedge \\ (\exists j \in \{1, \dots, k\}: f_j(\mathbf{x}^*) < f_j(\mathbf{x})) \end{aligned} \quad (2)$$

2.2 非支配解

如果 $\mathbf{x}^*$ 为非支配解,则满足式(3).

$$\neg \exists \mathbf{x} \in X: \mathbf{x} > \mathbf{x}^* \quad (3)$$

3 算法介绍

MSCC 框架主要包括目标函数的选取和多目标优化算法的选取.首先在 3.1 节介绍选取的目标函数,3.2 节介绍选取的多目标优化算法.

3.1 目标函数

Cai 等采用了两个最小化目标函数^[14]:

(1) 聚类目标函数

$$f_1(\mathbf{v}) = \sum_{j=1}^c \sum_{i=1}^N 2u_{ji}^m (1 - K(\mathbf{x}_i, \mathbf{v}_j)) \quad (4)$$

其中, $\mathbf{v}$ 为聚类中心组合 $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_c\}$, c 为聚类中心的个数,数值在实验中获得, N 为数据集的大小. $K(\mathbf{x}_i, \mathbf{v}_j)$ 为高斯核函数,计算公式如式(5)所示, u_{ji} 为个体 $\mathbf{x}_i$ 到聚类中心 $\mathbf{v}_j$ 的隶属度,计算过程如式(6)所示.

$$K(\mathbf{x}_i, \mathbf{v}_j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{v}_j\|^2}{\sigma^2}\right) \quad (5)$$

其中, $\sigma^2 = \frac{\max_{1 \leq i \leq N} \|\mathbf{x}_i - \bar{\mathbf{x}}\|^2}{\lambda}$, $\bar{\mathbf{x}}$ 为个体 $\mathbf{x}$ 的均值,在下面实验中获得 λ 的值.

$$u_{ji} = \frac{(1 - K(\mathbf{x}_i, \mathbf{v}_j))^{-1/(m-1)}}{\sum_{j=1}^c (1 - K(\mathbf{x}_i, \mathbf{v}_j))^{-1/(m-1)}} \quad (6)$$

式(4)和式(6)中的 m 一般取值为 2.

(2) 分类目标函数

$$f_2(\mathbf{v}) = \sum_{i=1}^N \delta(f(\mathbf{x}_i), \mathbf{y}_i) / N \quad (7)$$

其中 $\delta(\cdot, \cdot)$ 为判别函数,一般形式如式(8)所示:

$$\delta(x, y) = \begin{cases} 0, & x = y \\ 1, & x \neq y \end{cases} \quad (8)$$

$f(\mathbf{x}_i)$ 为分类函数,决定 $\mathbf{x}_i$ 属于哪个类别,一般形式如式(9)所示:

$$f(\mathbf{x}_i) = \arg \max_{1 \leq l \leq L} p(w_l | \mathbf{x}_i) \quad (9)$$

其中 w_l 为第 l 个类别, $p(w_l | \mathbf{x}_i)$ 为个体 $\mathbf{x}_i$ 为类别 w_l 的概率,计算公式如式(10)所示:

$$p(w_l | \mathbf{x}_i) = \sum_{j=1}^c p(c_j | \mathbf{x}_i) p(w_l | c_j) \quad (10)$$

其中 $p(c_j | \mathbf{x}_i)$ 为个体 $\mathbf{x}_i$ 为聚类 c_j 的概率,计算过程如式(11)所示, $p(w_l | c_j)$ 为聚类 c_j 为 w_l 类别的概率,计算过程如式(12)所示.

$$p(c_j | \mathbf{x}_i) = \frac{(2(1 - K(\mathbf{x}_i, \mathbf{v}_j)))^{-1}}{\sum_{j=1}^k (2(1 - K(\mathbf{x}_i, \mathbf{v}_j)))^{-1}} \quad (11)$$

其中 $\mathbf{v}_j$ 为第 j 个聚类中心.

$$p(w_l | c_j) = \frac{\text{Num}(\mathbf{x} \in w_l \text{ and } \mathbf{x} \in c_j)}{\text{Num}(\mathbf{x} \in c_j)} \quad (12)$$

3.2 多目标优化算法

在介绍多目标优化算法之前,首先介绍一下多目标优化算法种群中个体的选取.种群中个体由聚类中心向量构成,假设有 k 个聚类中心且分类数据的维数为 m ,则种群中的个体可以表示为 $\mathbf{x}_i = [x_{i1}, x_{i2}, \dots, x_{im}, \dots, x_{in}]$,其中 $n = k \times m$.

首先介绍 MOPSO^[15]优化 MSCC 框架的多目标问题的步骤.由于在优化过程中,非支配解个数较少,所以省去 MOPSO 划分超格和轮盘赌选择等操作.其中 MOPSO 粒子新位置计算步骤如算法 1.

算法 1:MOPSO 粒子新位置计算

```

V[i]: Pop[i]的速度;
BestPop[i]: Pop[i]当前最佳位置;
RepPop[i]:非支配解集的第 i 个粒子;
N:种群 Pop 大小;
Begin
  For i = 1:1: N
    计算 Pop[i]的速度 V[i];
  End
  For i = 1:1: N
    计算 Pop[i]的最新位置: Pop[i] = Pop[i] + V[i].
  End
End

```

其中,粒子速度的计算公式如下:

$$V[i] = w * V[i] + r_1 * (BestPop[i] - Pop[i]) + r_2 * (RepPop[r] - Pop[i]) \quad (13)$$

其中, w 为惯性权值,取值为 0.4, r_1 和 r_2 为分布在[0,1]的随机数, $RepPop[r]$ 为非支配种群 $RepPop$ 中随机选取的粒子.

算法 2:MOPSO 优化 MSCC 框架的多目标问题

```

N:种群 Pop 大小;
Begin
  iger = 1;
  初始化种群 Pop;
  For i = 1:1: N
    设定 Pop[i]的速度: V[i] = 0;
    计算 Pop[i]的目标函数值;
    初始化 Pop[i]的最佳位置: BestPop[i] = Pop[i];
  End
  从种群 Pop 中选取非支配解集 RepPop;
  While iger < max_ger
    For i = 1:1: N
      根据算法 1,计算 Pop[i]的新位置;
      计算 Pop[i]的目标函数值;
    End
    在 | Pop, RepPop | 中选取非支配解集 RepPop;
    For i = 1:1: N
      If Pop[i]支配 BestPop[i]
        更新 BestPop[i]: BestPop[i] = Pop[i];
      Else If BestPop[i]支配 Pop[i]
        保持 BestPop[i]不变;
      Else
        随机决定 BestPop[i]是否更新或保持;
      End
    End
    iger = iger + 1.
  End
End

```

NSGA-II^[16]的广泛应用源自于其采用的 Pareto 等级排序方法,以下首先介绍 NSGA-II 算法 Pareto 等级排序的计算步骤,如算法 3 所示;之后介绍 NSGA-II 优化 MSCC 框架中多目标问题的主要步骤,如算法 4 所示.

算法 3:NSGA-II 中 Pareto 等级排序

```

ni: Pop[i]被其余个体支配的次数;
Si: Pop[i]支配的个体集;
nrank: Pareto 等级;
N:种群 Pop 大小;
Begin
  For i = 1:1: N
    ni = 0; Si =  $\Phi$ ;
  End
  For i = 1:1: N - 1
    For j = i + 1:1: N
      If Pop[i]支配 Pop[j]
        ni = ni + 1; Si = Si  $\cup$  Pop[j];
      If Pop[j]支配 Pop[i]
        nj = nj + 1; Sj = Sj  $\cup$  Pop[i];
      End
    End
  End
  nrank = 1;
  While 种群 Pop 中仍有个体未分配 Pareto 等级
    将种群中 ni = 0 的 Pop[i]的 Pareto 等级设为 nrank, 将 Pop[i]支配的个体集 Si 中的 Pop[j]的被支配个数设为 nj - 1;
    nrank = nrank + 1;
    忽略种群中已分配 Pareto 等级的个体.
  End
End

```

算法 4:NSGA-II 优化 MSCC 框架的多目标问题

```

N:种群 Pop 大小;
Begin
  iger = 1;
  初始化种群 Pop;
  For i = 1:1: N
    计算 Pop[i]的目标函数值;
  End
  给种群 Pop 的个体分配非支配等级和拥挤度值;
  While iger < max_ger
    从种群 Pop 中选取子代种群 S;
    对子代种群 S 进行模拟二进制交叉和多项式变异等操作;
    计算种群 S 中个体的目标函数值;
    将 | Pop, S | 中个体分配非支配等级和拥挤度值, 经过选取更新得到新种群 Pop;
    iger = iger + 1.
  End
End

```

4 实验结果及其分析

本文实验所用数据集来自文献[17]和文献[18],如

表 1 所示,没有采用 Cai 等在实验中所用的 4 个数据集.其中, Lung_cancer, Pima_Indian_diabetes 和 Spambase 数据集的属性值有丢失,而 MSCC 框架并没有处理属性值丢失的机制;Ecoli 数据集中有两个类标只有两个个体,不适合本文的实验方法.

表 1 实验所用数据集

测试数据集	规模	类标数	实验数据维数
Lenses	24	3	4
Iris	150	3	4
WDBC	569	2	30
Soybean_small	47	4	35
Balance	625	3	4
Waveform	5000	3	21
Glass	214	6	9
Wine	178	3	13
Heart_disease	270	2	13
Sonar	208	2	60
Image_segmentation	2310	7	19
Vowel	528	11	10
Vehicle	846	4	18
Satimage	4435	6	36

在本文实验中, MOPSO 和 NSGA-II 的种群大小为 500,迭代次数为 100. NSGA-II 中模拟二进制交叉分布指数为 15,交叉概率为 0.9,变异分布指数为 20,变异概率为 $1/n$, n 为种群中个体的维数.

在和参考文献[14]中的算法进行对比时,为了更有说服力,本文算法中 (k, λ) 的取值和参考文献[14]中的参数相同,唯独 k 的取值不同,从参考文献[14]中的实验结果可以看出,几乎所有实验数据集都在 $k = 20$ 下得到了最优结果,在考虑到 $k = 100$ 时,聚类算法就失去了意义,所以本文 k 的取值范围取在了[1,20].

在每次独立实验中,从每个数据集随机选取一半数据作为训练样本集,其余为测试样本集.数据集的聚类中心的最小个数为该数据集的类别数,最大为 20, λ 取值集合为 {0.01,0.05,0.1,0.5,1,5,10,15}[14]. 为了给数据集选取合适的 (k, λ) ,对每个 (k, λ) 组合独立运行 30 次,求平均最大分类正确率,选取最大分类正确率最高的组合.实验结果如表 2 所示.

由表 2 可以看出,针对同样的测试数据集,不同的多目标优化算法得到平均最大分类正确率所用的参数也不同.从整体上可以看出,对于大多数问题,在 NSGA-II 优化作用下, MSCC 框架设计的分类器的最大分类正确率比 MOPSO 优化作用下的高.

NSGA-II 优化 MSCC 框架的多目标优化问题时,最优解集不仅保存有非支配解,也保留了大量的支配解.为了检查得到最大分类正确率的分类器对应的解是否为非支配解,本文对 NSGA-II 优化 MSCC 框架的多目标

问题进行测试, (k, λ) 取值如表 2 中所示,实验结果如表 3 所示.

表 2 (k, λ) 组合对应的最大分类正确率的均值和标准差

测试数据集	MOPSO			NSGA-II		
	k	λ	正确率	k	λ	正确率
Lenses	4	0.01	86.7 ± 11.9	3	5	26.67 ± 7.58
Iris	12	0.001	97.1 ± 1.7	17	1.0	98.04 ± 2.39
WDBC	2	0.1	97.3 ± 0.7	20	15	91.60 ± 2.07
Soybean_small	4	1	100 ± 0	5	0.01	100 ± 0
Balance	16	0.01	90.8 ± 1.2	17	0.1	91.38 ± 0.87
Waveform	13	0.01	81.92 ± 1.04	3	0.01	86.62 ± 0.62
Glass	20	0.01	68.9 ± 2.5	18	0.1	68.98 ± 3.84
Wine	6	0.1	98.3 ± 1.3	17	0.5	98.80 ± 1.16
Heart_disease	20	0.5	83.85 ± 2.56	18	0.05	86.49 ± 2.20
Sonar	18	0.01	85.6 ± 4.1	5	0.05	81.31 ± 2.34
Image_segmentation	19	0.05	57.83 ± 4.94	7	0.1	86.39 ± 1.75
Vowel	19	0.1	41.14 ± 0.78	20	0.01	41.99 ± 1.81
Vehicle	19	0.05	50.35 ± 5.95	15	0.01	53.52 ± 1.40
Satimage	8	0.1	69.63 ± 1.81	6	0.01	74.24 ± 2.19

表 3 最大分类正确率对应的 NSGA-II 解

测试数据集	NSGA-II
Lenses	非支配解
Iris	非支配解
WDBC	支配解
Soybean_small	非支配解
Balance	非支配解
Waveform	支配解
Glass	支配解
Wine	支配解
Heart_disease	支配解
Sonar	支配解
Image_segmentation	支配解
Vowel	支配解
Vehicle	非支配解
Satimage	支配解

从表 3 中可以看出,除了数据集 Lenses, Iris, Soybean_small, Balance_scale 和 Vehicle 外,在其余数据集上,都是支配解对应的分类器得到了最大分类正确率.多目标优化算法的目的是为了得到非支配解集,但是表 3 中支配解却得到了比非支配更高的分类正确率,为了弄清这个问题,本文又做了进一步分析.

多目标优化算法在一定的迭代次数内,种群会随着迭代次数的增加而优化为一个更好的解集(由支配解优化为非支配解,多样性提高,收敛性提高).为了观察最大分类正确率是否随着种群的优化而提高,本文设计了如下实验:在 0 ~ 100 迭代范围内,每隔 10 代,记录一次最大分类正确率,对每个数据集独立测试 10 次,求其均值.对文献[14]中 MOPSO 优化 MSCC 框架的多目标问题进行了实验.两种算法中, (k, λ) 取值如表 2

中所示,对每个数据集独立测试 10 次,实验结果如图 3 所示.

由图 3 可以看出,NSGA-II 优化 MSCC 框架的多目标问题时,在 14 个数据集中, Soybean_small, Waveform, Glass, Sonar, Image segmentation, Vowel, Vehicle 和 Satimage 随着种群的优化,平均最大分类正确率在不断提高,说明该优化过程有益于分类器的设计,而 WDBC, Balance 和 Wine 数据集在最初的优化过程中,平均最大分类正确率得到了提高,而后随着种群的优化,平均最大分类正确率却随之减小; Lense, Iris 和 Heart_disease 数据集随着种群的优化,最大分类正确率在不断减小.

同样,由图 3 可以看出, MOPSO 优化 MSCC 框架的多目标问题时,除了 Soybean_small, Sonar, Image segmentation, Vowel 和 Vehicle 数据集的平均最大分类正确率随着种群的优化而呈现上升趋势外,其余数据集的平均

最大分类正确率随着种群优化的变化趋势并不稳定.

因此,由图 3 可以得出如下结论:在处理部分数据集时, MSCC 框架设计的多分类器的性能并不随着多目标优化算法的种群的优化而提高,甚至会下降.

5 总结

本文首先介绍 Cai 等提出的 MSCC 框架,由于文献 [14]中 MOPSO 优化 MSCC 框架的多目标问题时,仅能得到少量非支配解,而在这种情况下, NSGA-II 由于采用了 Pareto 排序方法,在优化过程中能够保留大量优秀的支配解,这样有利于种群优化,所以本文引进 NSGA-II 优化 MSCC 框架的多目标问题.通过实验表明,在 NSGA-II 算法下, MSCC 框架设计的分类器的性能得到了一定提高;而对大部分数据集来说,最大分类正确率为 NSGA-II 种群中的支配解对应的分类器所得.最后发现

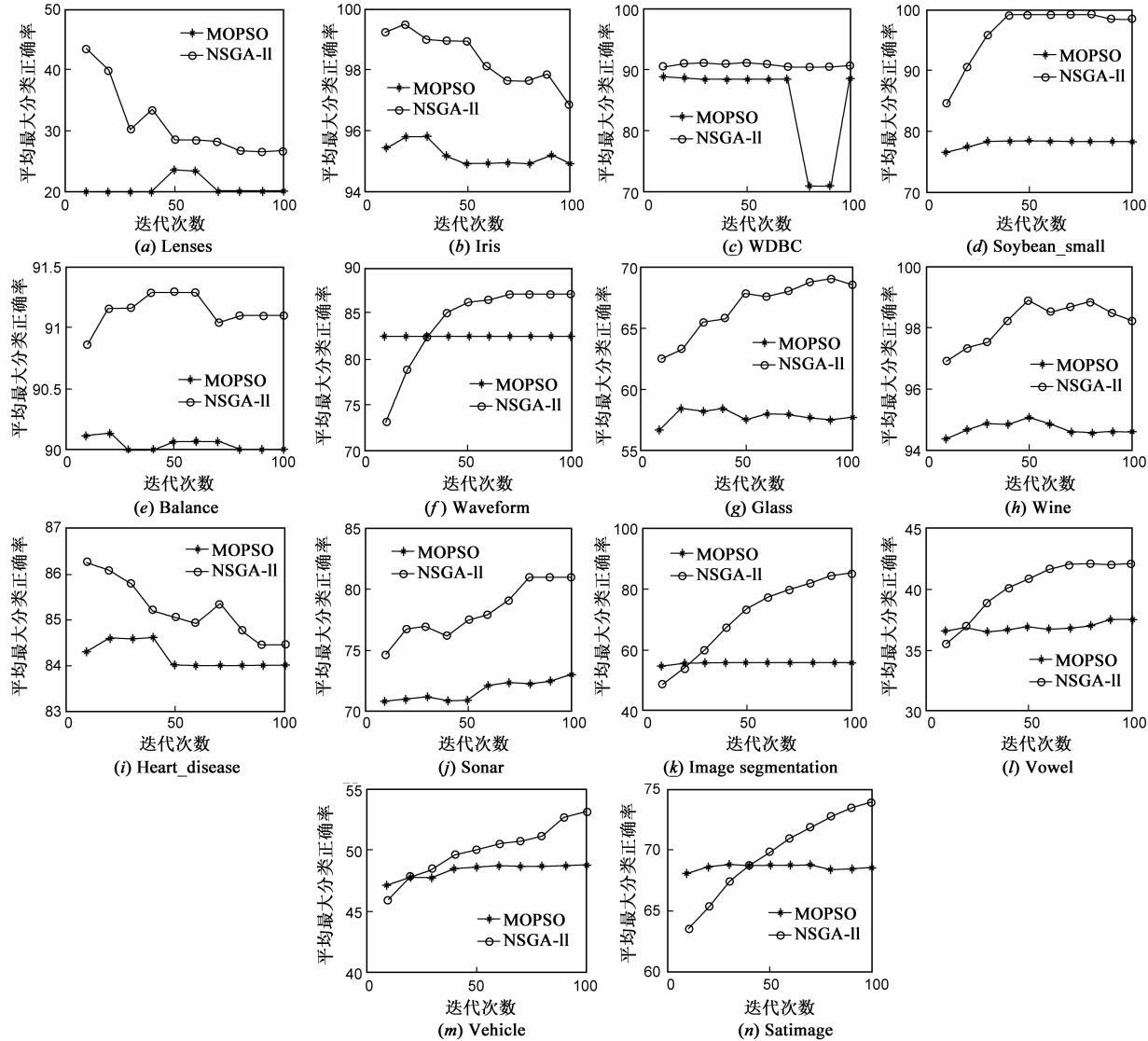


图3 MOPSO和NSGA-II优化MSCC框架的多目标问题,平均最大分类正确率随着迭代次数增加的变化曲线

对部分数据集,MSCC 框架设计的分类器的性能并不随着多目标优化算法的优化过程而提高,甚至还会下降.为何支配解对应的分类器比非支配解对应的分类器的性能好,分类器的性能会随着多目标优化算法的优化过程而下降,仍需进一步的研究.

参考文献

- [1] R Tagliaferri, A Staiano, D Scala. A supervised fuzzy clustering for radial basis function neural networks training[A]. Joint 9th IFSA World Congress and 20th NAFIPS International Conference[C]. USA: IEEE Press, 2001. 3: 1804 – 1809.
- [2] Y J Oyang, S C Hwang, Y Y Ou, C Y Chen, Z W Chen. Data classification with radial basis function networks based on a novel kernel density estimation algorithm[J]. IEEE Transactions on Neural Networks, 2005, 16(1): 225 – 236.
- [3] I Maglogiannis, H Sarimveis, C T Kiranoudis, A Chatziioannou, N Oikonomou, V Aidinis. Radial basis function neural networks classification for the recognition of idiopathic pulmonary fibrosis in microscopic images[J]. IEEE Transactions on Information Technology in Biomedicine, 2008, 12(1): 42 – 54.
- [4] 米爱中,郝红卫,郑雪峰,涂序彦.一种自整定权值的多分类器融合方法[J].电子学报,2009,37(11):2604 – 2609.
MI Ai-zhong, HAO Hong-wei, ZHENG Xue-feng, TU Xu-yan. A method of multiple classifier fusion with self-adjusting weights[J]. Acta Electronica Sinica, 2009, 37(11): 2604 – 2609. (in Chinese)
- [5] 连可,黄建国,王厚军,龙兵.一种基于遗传算法的 SVM 决策树多分类策略研究[J].电子学报,2008,36(8):1502 – 1507.
LIAN Ke, HUANG Jian-guo, WANG Hou-jun, LONG Bing. Study on a GA-based SVM decision-tree multi-classification strategy[J]. Acta Electronica Sinica, 2008, 36(8): 1502 – 1507. (in Chinese)
- [6] J Weston, C Watkins. Multi-class Support Vector Machines [R]. Technical Report CSD-TR-98-04, May 20, 1998: 1 – 10.
- [7] 李阳阳,石洪竺,焦李成,马文萍.基于流形距离的量子进化聚类算法[J].电子学报,2011,39(10):2343 – 2347.
LI Yang-yang, SHI Hong-zhu, JIAO Li-cheng, MA Wen-ping. Quantum-inspired evolutionary clustering algorithm based on manifold distance[J]. Acta Electronica Sinica, 2011, 39(10): 2343 – 2347. (in Chinese)
- [8] J A K Suykens, J Vandewalle. Least squares support vector machine classifiers[J]. Neural Processing Letters, 1999, 9(3): 293 – 300.
- [9] 李昆仑,黄厚宽,田盛丰.模糊多类 SVM 模型[J].电子学报,2004,32(5):830 – 832.
LI Kun-lun, HUANG Hou-kuan, TIAN Sheng-feng. Fuzzy support vector machine for multi-class classification[J]. Acta Elec-

tronica Sinica, 2004, 32(5): 830 – 832. (in Chinese)

- [10] 周伟达,张莉,焦李成.自适应支撑矢量机多用户检测[J].电子学报,2003,31(1):92 – 97.
ZHOU Wei-da, ZHANG Li, JIAO Li-cheng. Adaptive SVM for multi-user detection[J]. Acta Electronica Sinica, 2003, 31(1): 92 – 97. (in Chinese)
- [11] J R Quinlan. Induction of decision trees[J]. Machine Learning, 1996, 1(1): 81 – 106.
- [12] J R Quinlan. C4.5: Programs for Machine Learning[M]. San Mateo, Calif: Morgan Kaufmann, 1993.
- [13] J R Quinlan. Improved use of continuous attributes in C4.5[J]. Artificial Intelligence Research, 1996, 4: 77 – 99.
- [14] W L Cai, S C Chen, D Q Zhang. A multiobjective simultaneous learning framework for clustering and classification[J]. IEEE Transactions on Neural Networks, 2010, 21(2): 185 – 200.
- [15] C A C Coello, G T Pulido, M S Lechuga. Handling multiple objectives with particle swarm optimization[J]. IEEE Transactions on Evolutionary Computation, 2004, 8(3): 256 – 279.
- [16] K Deb, A Pratap, S Agarwal, T Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II[J]. IEEE Transactions on Evolutionary Computation, 2002, 6(2): 182 – 197.
- [17] C C Chang, C J Lin. LIBSVM: A library for support vector machines, 2001 [OL]. <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [18] C Blake, E Keogh, C J Merz. UCI Repository of Machine Learning Databases [OL]. Department of Information and Computer Science, <http://www.ics.uci.edu/~mlearn/ML-Repository.html>, 1998.

作者简介



尚荣华 女, 1979 年 11 月生于河北阜城. 2008 年毕业于西安电子科技大学电子工程学院, 获博士学位. 现为西安电子科技大学电子工程学院副教授. 主要研究方向为工程优化、智能信息处理等.

E-mail: rhshang@mail.xidian.edu.cn



胡朝旭 男, 1986 年 12 月生于河南洛阳. 2011 年 12 月毕业于西安电子科技大学电子工程学院, 获硕士学位. 主要研究方向为多目标优化、数据挖掘类等.

E-mail: 498175990@qq.com