# 一种蜜蜂交配优化聚类算法

罗 可,李 莲,周博翔
(长沙理工大学计算机与通信工程学院,湖南长沙 410114)

**摘 要:** K-means 算法因简单、高速等特点而被广泛应用,但该算法仍然存在依赖于初始聚类中心、易陷入局部最优等缺陷.为此,提出了一种蜜蜂交配优化聚类算法.该算法利用密度和距离初始化蜂群,并将局部搜索能力较强的粗糙集聚类算法作为工蜂的一种编码,以增强算法的局部搜索能力,最后在迭代过程中不断引入随机种群,增加种群的多样性,提高算法的全局寻优能力.实验结果表明,该算法不仅能有效抑制早熟收敛,而且具有较强的稳定性,较好的聚类效果.

**关键词:** 聚类;蜜蜂交配优化;粗糙集;K-means
**中图分类号:** TP18 **文献标识码:** A **文章编号:** 0372-2112 (2014)12-2435-07
**电子学报 URL:** http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2014.12.015

# A Honey-Bee Mating Optimization Clustering Algorithm

LUO Ke, LI Lian, ZHOU Bo-xiang
(*Institute of Computer and Communication Engineering, Changsha University of Sciences and Technology, Changsha, Hunan* 410014, *China*)

**Abstract:** K-means algorithm is the most widely used method due to its easy understanding and fast speed. However, this method has the disadvantage that the clustering results depend on the selection of the initial clustering center and it is easy to fall into local optimal. For this reason, this paper proposed a honey-bee mating optimization clustering algorithm. It generates initial swarm by density and distance, and regards rough set clustering algorithm which has strong local search ability as a code of the works to enhance the local search ability of the algorithm. At last, in order to improve the diversity level of the swarm and the global optimization ability of the algorithm, random swarm population are introduced continuously in the iterative process. Our experiments show that the proposed algorithm not only can effectively suppress premature convergence, but also has strong stability and produces good clustering results.

**Key words:** clustering; honey-bee mating optimization; rough set; K-means

## 1 引言

聚类就是将数据对象分成多个簇,同一个簇中对象之间具有较高的相似度,而不同簇中对象差别较大.聚类分析现已成为数据挖掘研究领域中一个非常活跃的研究课题[1].K-means 作为一种典型的聚类算法,因其依赖初始聚类中心、易陷入局部最优等缺陷,引起了广大学者的研究与改进.有的学者利用遗传算法[2]、差分演化[3]、PSO[4]、ACO[5]、人工蜂群[6,7]等智能优化算法改进 K-means 的不足,而有的则结合粗糙集[8],重力搜索[9]等对其进行改进,各种方法都取得了一定的效果,但对复杂问题,还存在精度不高等问题.

蜜蜂交配优化算法(Honey Bee Mating Optimization, HBMO)是 Abbass[10]于 2001 年提出的模拟蜜蜂繁殖行为的蜂群算法.因其全局搜索能力强,鲁棒性好等优点得到了广泛的研究与应用.Arit Thammano 等人提出了一种蜂王数自组织以及多个蜂巢的蜜蜂交配算法[11].Poolsamran P 等人提出了一种实数编码的蜂王数自组织的蜜蜂交配算法[12].孟伟等人将蜂群交配思想用于 GA 算法,提高了 GA 算法的全局收敛性[13].Fathian 等人将 HBMO 用于聚类分析[14].这些算法各有特色,但无法保证全局探索和局部搜索能力的平衡,易早熟收敛.因此,本文提出了一种新的蜜蜂交配算法.该算法从蜂群初始化、工蜂编码方法、随机蜜蜂的引入三个方面改进了传统 HBMO 算法的不足,平衡了全局探索和局部搜索能力,增强了算法的综合性能.本文将局部搜索能力强的

收稿日期:2013-07-15;修回日期:2014-03-20;责任编辑:梅志强
基金项目:国家自然科学基金(No.11171095,No.71371065);湖南省自然科学衡阳联合基金(No.10JJ8008);湖南省科技计划项目(No.2013SK3146)

粗糙集聚类算法和全局寻优能力强的新 HBMO 结合，提出了一种蜜蜂交配优化聚类算法(用 HBMO-RK 表示)，并通过实验证明了新算法的有效性.

## 2 粗糙集聚类算法

### 2.1 K-means 聚类算法简介

K-means 算法从给定样本集中找到 $k$ 个聚类中心 $\{a_1, a_2, \cdots, a_k\}$，按最小距离原则将所有样本分配到对应的类 $C_i$ 中，从而将样本集划分为 $k$ 个簇 $C_1, C_2, C_3, \cdots, C_k$；按 $C_i = \frac{1}{|C_i|}\sum_{x\in C_i} x$ 更新聚类中心，其中 $|C_i|$ 为第 $i$ 个簇的样本数，再按最小距离原则更新样本的所属类，根据使函数 $E = \sum_{i=1}^{k}\sum_{x\in C_i} |x - a_i|^2$ 最小准则，迭代至簇中心不变[15].

### 2.2 粗糙集理论

粗糙集理论[16]主要研究不精确和模糊的知识，在数据挖掘领域得到了成功应用. 下面给出粗糙集中与本文相关的一些定义.

**定义 1(上近似、下近似及边界集)** 给定知识库 $K=(U, R)$，对 $X\neq\phi$ 且 $X\subseteq U$，一个等价关系 $R\in \mathrm{ind}(K)$. 称 $\underline{R}(X) = \bigcup\{Y\in U/R \mid Y\subseteq X\}$ 为 $X$ 关于 $R$ 的下近似. 称 $\bar{R}(X) = \bigcup\{Y\in U/R \mid Y\cap X\neq\phi\}$ 为 $X$ 关于 $R$ 的上近似，而 $\mathrm{BNR}(X) = \bar{R}(X) - \underline{R}(X)$ 则称为 $X$ 的 $R$ 边界集.

**定义 2(粗糙集)** 若 $\bar{R}(X)\neq\underline{R}(X)$，则 $X$ 为 $R$ 的粗糙集，否则称 $X$ 为 $R$ 精确集.

### 2.3 粗糙集聚类

#### 2.3.1 适应度函数

用类内距离来评价聚类的内聚程度，如式(1)：

$$J_w = \sum_{i=1}^{k}\left(\omega_l * \sum_{x_j\in \underline{c}_{il}} \| x_j - v_i \|^2 + \omega_{\mathrm{bnr}} * \sum_{x_j\in c_{i\mathrm{bnr}}} \| x_j - v_i \|^2\right) \tag{1}$$

其中：$v_i$、$\underline{c}_{il}$、$c_{i\,\mathrm{bnr}}$ 分别为第 $i$ 类的聚类中心，下近似集和边界集，$\omega_l$，$\omega_{\mathrm{bnr}}$ 分别为第 $i$ 类的下近似集和边界集的权重.

适应度函数表示如式(2)，其中 $i$ 表示第 $i$ 只蜜蜂.

$$F_i = 1/J_w \tag{2}$$

#### 2.3.2 新的聚类中心

$$v_j = \begin{cases} \dfrac{\omega_l}{|\underline{c}_{jl}|}\sum_{x_i\in\underline{c}_{jl}} x_i + \dfrac{\omega_{\mathrm{bnr}}}{|c_{j\,\mathrm{bnr}}|}\sum_{x_i\in c_{j\mathrm{bnr}}} x_i, & c_{j\mathrm{bnr}}\neq\phi, \text{否则} \\ \dfrac{\omega_l}{|\underline{c}_{jl}|}\sum_{x_i\in\underline{c}_{jl}} x_i \end{cases} \tag{3}$$

其中 $j = 1, 2, \cdots, k$，$|\underline{c}_{jl}|$、$|c_{j\mathrm{bnr}}|$ 分别表示下近似集和边界集的样本个数.

#### 2.3.3 算法步骤

**Step1** 确定初始聚类中心 $v_1, v_2, \cdots, v_k$，其中 $k$ 为聚类数目.

**Step2** 将每个样本 $x_i$ 根据近邻原则分配给最近的类的上下近似集. 即对 $\forall\, x_m\in U$，找出与其距离最小的中心点 $v_i$，有 $d(x_m, v_i) = \mathrm{Min}\{d(x_m, v_j), j = 1, 2, \cdots, k\}$，则 $x_m\in\bar{v}_i$；如果 $\exists\, c_j$，使得 $d(x_m, v_i) - d(x_m, v_j) < \gamma * d(v_i, v_j)$，则令 $x_m\in\bar{v}_j$，否则 $x_m\in\underline{v}_i$.

**Step3** 根据式(2)计算适应度，如果 $\| F(t) - F(t-1) \| \leqslant\varepsilon$ 或者 $t\geq t_{\max}$，则结束；否则根据式(3)更新聚类中心，且 $t = t + 1$，转 Step2.

## 3 蜜蜂交配优化算法

一个完整的蜂群由蜂王、雄蜂、工蜂、幼蜂组成. 蜂王是蜂群中唯一能繁殖后代的雌蜂，是工蜂从幼蜂中精心培养出来的. 蜂王的主要任务是与不同的雄蜂交配与产卵. 雄蜂主要职责是与蜂王交配，便随之死亡. 工蜂负责照顾幼蜂、采蜜等工作. 蜜蜂交配算法的步骤如下[10]：

**Step1** 随机初始化一个蜂群，蜂群中适应度值最大的个体为蜂王，其余的为雄蜂.

**Step2** 婚飞：首先初始化蜂王的速度 $S(0)$ 和能量 $E(0)$，在婚飞过程中，雄蜂以概率 $p_i$(式(4))与蜂王交配，如果交配成功，则将雄蜂的染色体存入蜂王的受精囊，雄蜂死亡. 按式(5)、(6)分别更新蜜蜂的速度和能量. 直到受精囊满或者蜂王的能量达到预先设定的临界值.

$$p_i = \mathrm{e}^{-\Delta(F)/s(t)} \tag{4}$$

$$S(t+1) = \mu * S(t) \tag{5}$$

$$E(t+1) = E(t) - \theta \tag{6}$$

其中 $\Delta(F)$ 为雄蜂与蜂王的适应度之差，$s(t)$、$E(t)$ 为 $t$ 时刻蜂王的速度和能量；$\mu\in(0,1)$ 为衰减系数，$\theta\in(0,1)$ 为每次交配后能量的减少量.

**Step3** 繁殖过程：从受精囊中随机选择一个染色体与蜂王的染色体交叉，产生幼蜂，然后对幼蜂进行变异操作，以提高解的多样性，直到幼蜂数量达到设定值.

**Step4** 饲养过程：工蜂代表不同的启发式算法，利用工蜂进一步提高幼蜂的性能.

**Step5** 如果适应度最大的幼蜂优于蜂王，则将其替代蜂王，否则蜂王不变，其余幼蜂为雄蜂.

**Step6** 判断是否达到最大婚飞次数，如果达到，则停止，否则转至 Step2.

## 4 改进蜜蜂交配的聚类算法

### 4.1 改进思想

(1)蜂群初始化的改进

传统的蜜蜂交配算法随机选取初始蜂群，很难保

证优良蜜蜂的存在,且蜂群在备选解空间中分配不均,也影响了算法的整体性能.鉴于此,本文提出了一种基于密度和最大最小距离的蜂群初始化方法,步骤如下:

**Step1**　计算任意两个样本间距离 $\mathrm{dist}(x_i,x_j)$,记录于 $\boldsymbol{D}$ 中,则样本间平均距离为 $\bar{d}=\sum \mathrm{dist}(x_i,x_j)/n^2$, $i,j=1,2,\cdots,n$.

**Step2**　样本 $x_i$ 的密度定义为 $\mathrm{Density}(x_i)=\{p\in C \mid \mathrm{dist}(x_i,p)\leqslant r\}$,表示以 $x_i$ 为中心点,$r$ 为半径组成的球体中包含的样本数.其中 $r=\alpha*\bar{d}$,$\alpha$ 为常数,$C$ 为样本集.则样本集平均密度为 $\bar{\rho}(x_i)=\sum_{i=1}^{n}\mathrm{Density}(x_i)/n$.根据 $\mathrm{Density}(x_i)\leqslant\bar{\rho}(x_i)/4$ 将孤立点从 $C$ 中排除.

**Step3**　基于最大最小距离法得到 $k$ 个初始聚类中心:选取密度最高的样本为第一个聚类中心 $v_1$,离其距离最大的样本为第二个聚类中心 $v_2$,对于 $C$ 中剩余样本,根据矩阵 $D$,分别求出其中心到 $v_1,v_2,\cdots,v_m$ 距离为 $d_{i1},d_{i2},\cdots,d_{im}$,取 $d_i=\mathrm{Min}(d_{i1},d_{i2}\cdots,d_{im})$,$d=\mathrm{Max}(d_i)$对应的样本为第 $i$ 个聚类中心 $v_i$,以此类推计算 $v_k$.得到第一个染色体(候选解).反复执行 $N+1$ 次,生成含 $N+1$ 个染色体$\{Z_1,Z_2,\cdots,Z_{N+1}\}$.

**Step4**　按式(2)计算每个染色体的适应度,适应度最大者为蜂王 $q$,其他的为雄蜂$\mathrm{dr}_i$.

实验证明,采用这种方法生成的初始蜂群性质优良,为后续的寻优奠定了更好的基础.

(2)工蜂编码方法

传统的蜜蜂交配算法及其改进算法将工蜂编码成不同的启发式函数来进一步改进蜂王和幼蜂的基因型,如随机翻转[10]、随机更新[10]、单点交换[11]、两点交换[11].但这些启发式函数都是随机变换或者交换蜂王或在幼蜂染色体中的基因,随机性较强,局部寻优能力较差,影响了算法的整体性能.考虑到粗糙集聚类算法具有较强的局部搜索能力,因此,本文将工蜂编码成粗糙集聚类算法来饲养蜂王和新生的幼蜂,能快速有效地提高种群的性能,增强算法的局部搜索能力.具体的工蜂编码方法如下:

在本文中工蜂代表粗糙集聚类算法,因此本文用聚类中心来编码工蜂.每只工蜂用其所饲养的蜂王或幼蜂的染色体以及适应度值来具体编码,如式(7)所示.工蜂对蜂王或幼蜂的饲养过程则是以蜂王或幼蜂染色体所代表的聚类中心为初始聚类中心进行粗糙集聚类得到新的聚类中心,如果新解的适应度值大于旧解的适应度值,则用新解代替旧解来改变蜂王或者幼蜂的基因型,具体饲养过程如图 1 所示.其中 $z_{i1},z_{i2},\cdots,z_{ik}$表示 $k$ 个聚类中心,它代表蜜蜂的染色体.

$$G_i=(z_{i1},z_{i2},\cdots,z_{ik},F_i) \tag{7}$$

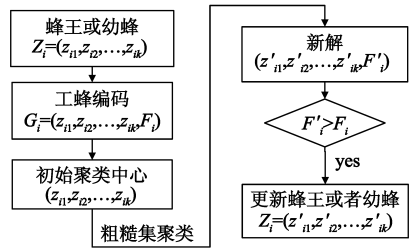



图1　工蜂饲养过程图

(3)引入随机蜜蜂

传统的蜜蜂交配算法,由于幼蜂都从蜂王继承基因,子代种群向最优解进化的可能性变大,另一方面,算法陷入局部最优解的可能性也加大了.为了避免早熟,本文在迭代过程中不断引入随机蜜蜂来替代性能较差的雄蜂,来保持进化种群的多样性,加强算法的全局寻优能力.具体思想如下:

对每只雄蜂设置一个计数器$\mathrm{tag}_i$,记录每只雄蜂在蜂王婚飞过程中未与蜂王交配成功的次数.如果蜂王在婚飞过程中与第 $i$ 只雄蜂成功交配,则$\mathrm{tag}_i=0$,否则$\mathrm{tag}_i++$.若$\mathrm{tag}_i$ 等于预先设定的阈值 Lim,则说明此雄蜂性能较差,且已经陷入局部最优解而不能跳出,根据式(8)随机搜索一个新解替代第 $i$ 只雄蜂,否则不变.

$$z_i^j=z_{\mathrm{Min}}^j+\phi(z_{\mathrm{Max}}^j-z_{\mathrm{Min}}^j) \tag{8}$$

其中 $j\in\{1,2,\cdots,\zeta\}$,$z_{\mathrm{Min}}^j$、$z_{\mathrm{Max}}^j$表示样本中第 $j$ 维的最小值和最大值.

改进的蜜蜂交配优化算法的代进化过程如图 2 所示.

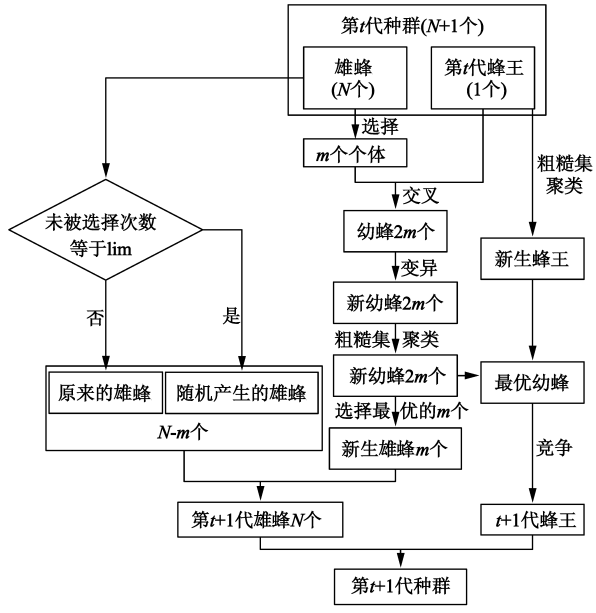



图2　代进化过程示意图

## 4.2　算法设计

(1)蜜蜂编码

每一个染色体代表一个问题的候选解,组成染色体的每个基因代表候选解的一个参数.本文采用实数编码,每个染色体由 $k$ 个聚类中心 $z_{ik}$ 组成,每个基因代表一个聚类中心的一维,则蜜蜂染色体编码为 $Z_i = (z_{i1}, z_{i2}, \cdots, z_{ik})$.如果 $z_{i1} = (2,3,1)$, $z_{i2} = (5,4,2) \cdots z_{ik} = (4,6,3)$,则具体编码如图 3 所示.

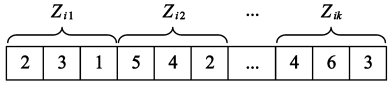



**图3 染色体编码**

(2)算法步骤

**Step1** 蜂王与雄蜂初始化.

**Step2** 婚飞.初始化蜂王的能量和速度,判断受精囊是否满或者蜂王的能量是否为预先设定的临界值,如果是,则转入 Step3;否则随机选择一个雄蜂,根据式(4)计算它的交配概率 $p_i$,如果 $p_i > r$(随机数 $r \in [0,1]$),则交配成功,将它的染色体加入受精囊中,雄蜂死亡,$\text{tag}_i = 0$;否则$\text{tag}_i + +$.根据式(5)、(6)更新蜂王的速度和能量.式(6)中,$\theta = e * E(0)/\text{sc}$,sc 为受精囊容量,$E(0)$为初始能量,$e = 0.5$.

**Step3** 繁殖过程.蜂王 $q$ 与每只雄蜂$\text{dr}_i$ 的交叉操作按式(9)、(10)进行并产生两只新幼蜂$\text{br}_i$、$\text{br}_i'$,$\beta \in (0,1)$.

$$\text{br}_i = (1-\beta) * \text{dr}_i + \beta * q \tag{9}$$

$$\text{br}_i' = (1-\beta) * q + \beta * \text{dr}_i \tag{10}$$

**Step4** 饲养过程.利用工蜂对蜂王和幼蜂局部搜索产生新解,如果新解的适应度值大于旧解的适应度值,则用新解代替旧解.

**Step5** 随机搜索外来蜂.如果第 $i$ 只雄蜂的计数器 $\text{tag}t_i \geq \text{lim}$,则根据式(8)随机搜索一个新解代替它,否则不变.

**Step6** 生成新种群.如果适应度最大的幼蜂优于蜂王,则将其与蜂王交换,否则不变.父亲相同的两只幼蜂中选择最优的一只替代已死亡的父亲(雄蜂).

**Step7** 若当前迭代次数达到最大迭代次数,则停止迭代;否则转到 Step2,且 $T = T + 1$.

## 5 实验结果与分析

实验环境:软件:操作系统 Windows XP,集成开发环境:Microsoft Visual C + + 6.0;MATLAB 7.0.硬件:Intel(R) Core(TM) i3-2100 CPU @3.10GHz,4GB 的内存.

为验证蜜蜂初始化改进的有效性以及随机蜜蜂引入的有效性,本文引入了两种对比算法,一种是采用随机初始化方法得到的算法(简称 HBMO-RK-RI),另一种是不引入随机蜜蜂的算法(简称 HBMO-RK-URB),这两种算法其他方面与本文算法(HBMO-RK)一致,且比较了本文算法(HBMO-RK)、HBMO-RK-RI、HBMO-RK-URB 三种算法在 Iris 和 Wine 标准数据集上的测试结果.

为验证本文算法的综合性能,实验中将其在 Iris 和 Wine 标准数据集上的测试结果与一些非启发式聚类算法和一些启发式聚类算法相比较.其中非启发式聚类算法有:Km、PCA-K、LLE-K、LDA-K[17]、Res-K[18]、SC[19]、RK、F-RK、UD-RK[20];启发式聚类算法有:GA、K-NM-PSO[4]、ACO[5]、ABC1[6]、ABC2[7]、HBMO[14]、PSO-ACO-K[21].各数据集的特征如表 1 所示.经过 10 次实验,本文算法效果最好时各数据集中各参数的设置如下:

蜂王数 QN = 1,雄蜂数 $N = 100$,受精囊容量 sc = 80,最大迭代次数 $I = 1000$, $E(0) = 0.8$, EM = 0.1, $S(0) = 0.8$, $\mu = 0.98$, Lim = 4. Iris 中参数选择:$\omega_l = 0.72$, $\omega_{\text{bnr}} = 0.28$, $a = 0.5$, $\gamma = 0.2$. Wine 中参数选择:$\omega_l = 0.7$, $\omega_{\text{bnr}} = 0.3$, $a = 0.09$, $\gamma = 0.3$.

**表 1 实验中涉及的数据集**

| 数据集名称 | 样本数目 | 属性维数 | 类别数 |
|---|---|---|---|
| Iris | 150 | 4 | 3 |
| Wine | 178 | 13 | 3 |

在 Iris、Wine 数据集上,本文算法(HBMO-RK)、HBMO-RK-RI、HBMO-RK-URB 三种算法的适应度随迭代次数增加的变化情况分别如图 4、图 5 所示.因 600 到 1000 次迭代的适应度值均不变,所以本文只画了前 600 次迭代的适应度图,以便更清楚观察适应度的变化情况.

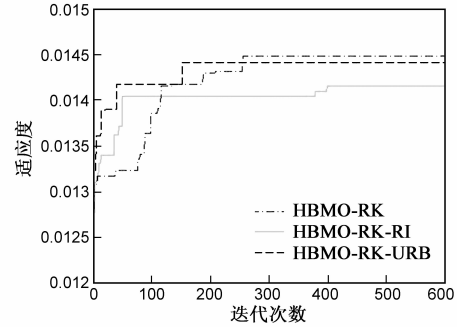



**图4 Iris数据集上适应度变化情况**

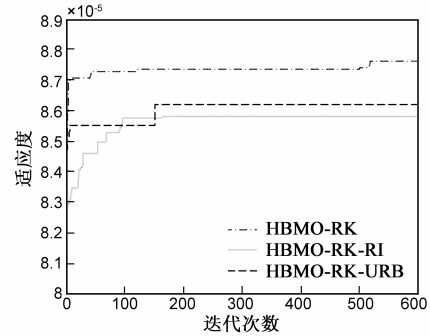


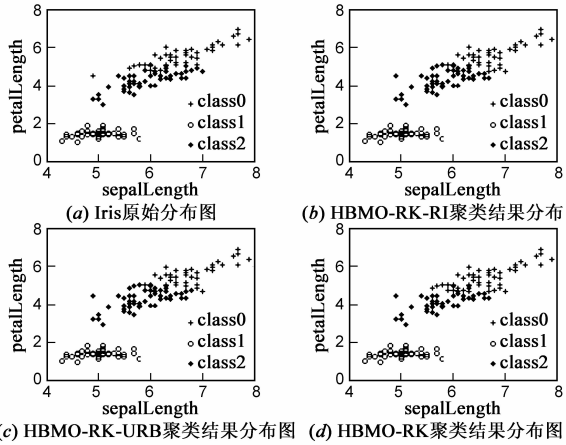

**图5 Wine数据集上适应度变化情况**

由图 4、图 5 可知:在 Iris 中,HBMO-RK 适应度初值为 0.0124699,经过后期一系列的搜索,算法最终达到全局最优值为 0.0144759,比初值提高了 16.1%.在 Wine 中,HBMO-RK 适应度初值为 8.16664e-005,算法最终达到全局最优值为 8.76188e-005,比初值提高了 7.4%.由此可见,采用密度和距离初始化方法使得算法初始解较好,提高了算法搜索起点,缩短了适应度变化范围,为后续的寻优奠定了更好的基础.Iris 和 Wine 中 HBMO-RK 与 HBMO-RK-URB 算法的适应度初值相等,但大于 HBMO-RK-RI 算法,在 Iris 中比 HBMO-RK-RI 算法大 3.6%,在 Wine 中比 HBMO-RK-RI 算法大 1.8%,且全局最优解 HBMO-RK > HBMO-RK-URB > HBMO-RK-RI,说明了,采用密度和距离初始化方法的有效性.由于 HBMO-RK 算法初始解较优,且新的工蜂编码方法具有较强的局部搜索能力,因此其能快速收敛于局部最优,而随着迭代次数的增加,不断引入随机蜜蜂,保持进化种群多样性,使算法能多次跳出局部最优,最终达到全局最优.与其对比,HBMO-RK-URB 算法则因在迭代中未引入随机蜜蜂,种群较单一,致使得到的全局最优解要比 HBMO-RK 算法差.由此可见,本文算法能有效地避免陷入局部最优值,具有较强的全局搜索能力.

同时,为了表现本文算法的聚类效果,本文分别选取 Iris、Wine 数据集中的两维特征进行描述.HBMO-RK、HBMO-RK-RI、HBMO-RK-URB 三种算法的聚类分布如图 6 和 7 所示.

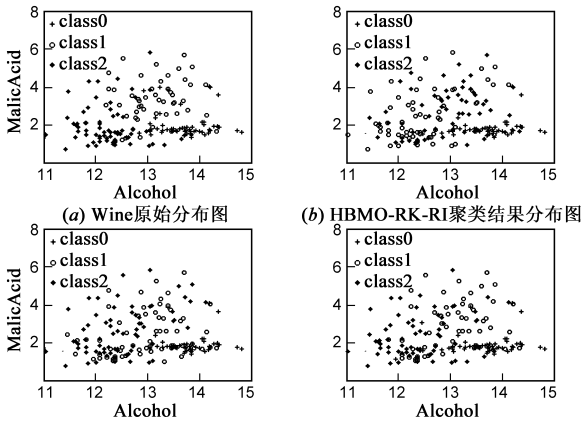
从图 6 到 7 可以看出,聚类效果 HBMO-RK > HBMO-RK-URB > HBMO-RK-RI,本文算法聚类准确率最高.由此可见,本文中三大创新点的引入增强了算法的综合性能.由图 7 可知,粗糙集的引入使得 HBMO-RK 算法对边界数据处理能力较强,聚类结果与原始数据样本分布比较接近.

在 10 次独立实验中,每次都会产生不同的随机种子来得到应用的随机参数值.将 10 次运行结果求平均,得到的实验结果如表 2 ~ 表 4 所示.



(*a*) Iris原始分布图　(*b*) HBMO-RK-RI聚类结果分布图
(*c*) HBMO-RK-URB聚类结果分布图　(*d*) HBMO-RK聚类结果分布图

**图6　Iris中聚类结果分布图**



(*a*) Wine原始分布图　(*b*) HBMO-RK-RI聚类结果分布图
(*c*) HBMO-RK-URB聚类结果分布图　(*d*) HBMO-RK聚类结果分布图

**图7　Wine中聚类结果分布图**

**表 2　各非启发聚类算法聚类准确率与本文算法比较**

| | Km | PCA-K | LLE-K | LDA-K | Res-K | SC | RK | F-RK | UD-RK | HBMO-RK |
|---|---|---|---|---|---|---|---|---|---|---|
| Iris% | 78.2 | 88.67 | 77.33 | 98.00 | 96.67 | 84.67 | 87.11 | 89.79 | 92.13 | 93.33 |
| Wine% | 52.1 | 65.23 | 65.87 | 69.66 | 69.66 | 70.79 | 64.62 | 72.37 | 73.37 | 83.15 |

**表 3　各启发式聚类算法在 Iris 数据集中的聚类结果比较**

| 算法 | 类内距离 | | | 函数评估 | 平均 |
|---|---|---|---|---|---|
| | 最小 | 平均 | 最大 | 次数 | 准确率 |
| Km | 97.333 | 106.050 | 120.450 | 120 | 78.20 |
| GA | 113.987 | 125.197 | 139.778 | 38128 | 77.80 |
| ACO | 97.101 | 97.172 | 97.808 | 10998 | 77.90 |
| K-NM-PSO | 96.660 | 96.670 | 97.010 | 4556 | 89.93 |
| PSO-ACO-K | 96.650 | 96.650 | 96.650 | 2480 | 78.80 |
| HBMO | 96.752 | 96.953 | 97.758 | 11214 | 78.10 |
| ABC1 | 96.650 | 96.650 | 96.650 | — | 89.80 |
| ABC2 | 78.940 | 78.940 | 78.940 | 8658 | — |
| HBMO-RK-RI | 70.655 | 75.994 | 78.694 | 40299 | 90.67 |
| HBMO-RK-URB | 69.396 | 72.456 | 77.103 | 15453 | 91.33 |
| HBMO-RK | 69.080 | 71.552 | 76.556 | 25856 | 93.33 |

**表 4　各启发式聚类算法在 Wine 数据集中的聚类结果比较**

| 算法 | 类内距离 | | | 函数评 | 平均 |
|---|---|---|---|---|---|
| | 最小 | 平均 | 最大 | 估次数 | 准确率 |
| Km | 16555.68 | 18061.00 | 18563.12 | 390 | 52.10 |
| GA | 16530.534 | 16530.534 | 16530.534 | 33551 | 51.50 |
| ACO | 16530.534 | 16530.534 | 16530.534 | 9306 | 51.90 |
| K-NM-PSO | 16292.000 | 16293.00 | 16279.46 | 46459 | 71.91 |
| PSO-ACO-K | 26295.310 | 26295.310 | 26295.310 | 6315 | 52.10 |
| HBMO | 16357.284 | 16357.284 | 16357.284 | 7238 | 51.80 |
| ABC1 | 16292.180 | 16292.870 | 16294.170 | — | 72.40 |
| ABC2 | 16257.280 | 16260.520 | 16279.460 | 17554 | — |
| HBMO-RK-RI | 11656.194 | 11662.100 | 11699.600 | 16766 | 72.47 |
| HBMO-RK-URB | 11601.688 | 11607.875 | 11643.143 | 15453 | 78.65 |
| HBMO-RK | 11413.100 | 11481.700 | 11521.000 | 52419 | 83.15 |

由表 2 可知:在 Iris 数据集中,本文算法聚类准确率稍低于 LLE-K 和 Res-K 算法,却远高于其他 7 种聚类算法,比 Km 高出了 19.35 个百分点.在 Wine 数据集中,

本文算法聚类准确率最高为 83.15%,比次高的 UD-RK 算法提高了 13.32%.因此,本文算法引入蜜蜂交配优化算法思想具有较好的聚类效果,聚类准确率高于多数非启发式聚类算法,在 Wine 数据集中表现尤为突出.

通过比较表 3 和表 4 各项实验结果可知:本文算法类内距离、准确率都优于 HBMO-RK-RI 以及 HBMO-RK-URB,说明了蜜蜂初始化改进的有效性以及随机蜜蜂引入的有效性.相比其他算法,本文算法的聚类准确率有很大的提高,在 Iris 数据集中,比最优的 K-NM-PSO 算法高出 3.78%.在 Wine 数据集中,比最优的 ABC1 算法高出 14.85%.说明了采用粗糙集聚类编码工蜂,具有较强的局部搜索和边界处理能力,能使蜜蜂找到聚类效果较好的解,表明本文算法的聚类效果远好于其他几种算法.此外,本文算法的最大、最小、平均类内距离均远远小于其他几种算法,在 Wine 数据集上表现尤为明显,比 ABC2 算法减小了 29.39%.说明本文算法稳定性得到了很大提高.函数评估次数表示算法收敛于最优解时适应度函数执行的次数,用来评价算法的收敛性.Km 的函数评估次数最小,其他几种算法因为加入智能优化算法,每次迭代中评价次数增大,致使总体的评价次数远大于 Km.而几种智能优化算法在 Iris 数据集中,GA 收敛速度最差,在 33551 次评价后到达最优值 125.197.PSO-ACO-K 最快,在 2480 次评价内达到了最优值 96.650.在 Wine 数据集中,PSO-ACO-K 收敛速度最快,在 6315 次评价内达到了最优值 26295.310.本文算法因在迭代中不断引入随机蜜蜂,多次跳出局部最优解,致使函数评价次数较大 52419,但得到的最优值 11413.100 远远小于 PSO-ACO-K 算法,说明了本文算法虽然收敛速度慢,类内距离和聚类准确率却得到了很大的改进.虽然其他几种算法收敛速度优于文本算法,但没能多次跳出局部最优,最终达不到全局最优解,致使算法的准确率远低于本文算法.

## 6 结束语

本文提出了一种改进的蜜蜂交配优化算法,并将该算法用于聚类中.采用密度和最大最小距离法初始化蜂群,得到适应度较大的蜜蜂,为后续的寻优奠定了更好的基础;将粗糙集聚类算法作为工蜂的一种编码来饲养蜂王和新生的幼蜂,增强了算法的局部搜索能力;迭代过程中不断引入随机蜜蜂,增加了种群的多样性,使算法能有效地跳出局部最优而找到全局最优解,提高了算法的全局寻优能力.实验结果表明本文算法全局寻优能力强,具有较高的准确率和较强的稳定性.当然,本文也还存在着一些不足:收敛速度慢,如何提高收敛速度,将成为我们下一步的研究方向.

## 参考文献


[1] Han J W, Kamber M. Data Mining: Concepts and Techniques [M]. New York: Morgan Kaufmann Publishers, 2001. 335 – 388.

[2] Xiao J, Yan Y P, et al. A quantum inspired genetic algorithm for K-means clustering[J]. Expert Systems with Applications, 2010, 37(7): 4966 – 4973.

[3] Kwedlo W. A clustering method combining differential evolution with the K-means algorithm[J]. Pattern Recognition Letters, 2011, 32(12): 1613 – 1621.

[4] Kao Y T, Zahara E. A hybridized approach to data clustering [J]. Expert Systems with Applications, 2008, 34(3): 1754 – 1762.

[5] Shelokar P S, Jayaraman V K. An ant colony approach for clustering[J]. Analytica Chimica Acta, 2004, 509(2): 187 – 195.

[6] Zhang C S, Ouyang D T, Ning J X. An artificial bee colony approach for clustering[J]. Expert Systems with Applications, 2010, 37(7): 4761 – 4767.

[7] Anan B, Booncharoen S, Tiranee A. The best-so-far ABC with multiple patrilines for clustering problems[J]. Neurocomputing, 2013, 116: 355 – 366.

[8] Chen J Y, Zhang C S. Efficient clustering method based on rough set and genetic algorithm[J]. Procedia Engineering, 2011, 15: 1498 – 1503.

[9] Abdolreza H, Salwani A. A combined approach for clustering based on K-means and gravitational search algorithms[J]. Swarm and Evolutionary Computation, 2012, 6: 47 – 52.

[10] Abbass H A. MBO: marriage in honey bees optimization: a haplometrosis polygynous swarming approach[A]. Proceedings Congress on Evolutionary Computation 2001[C]. Seoul: IEEE Service Center, 2001. 207 – 214.

[11] Thammano A, Poolsamran P. SMBO: A self-organizing model of marriage in honey-bee optimization[J]. Expert Systems with Applications, 2012, 39(5): 5576 – 5583.

[12] Poolsamran P, Thammano A. A modified marriage in honey-bee optimization for function optimization problems[J]. Procedia Computer Science, 2011, 6: 335 – 342.

[13] 孟伟,韩学东,洪炳.蜜蜂进化型遗传算法[J].电子学报,2006,34(7):1294 – 1300.
Meng W, Han X D, Heng B. Bee evolutionary genetic algorithm[J]. Acta Electronica Sinica, 2006, 34(7): 1294 – 1300. (in Chinese)

[14] Fathian M, Amiri B. Application of honey-bee mating optimization algorithm on clustering[J]. Applied Mathematics and Computation, 2007, 190(2): 1502 – 1513.

[15] Macqueen J. Some methods for classification and analysis of multivariate observations[A]. Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability[C].

Berkeley: University of California Press, 1967. 281 – 297.
[16] 王彪,段禅伦,等. 粗糙集与模糊集的研究及应用[M]. 北京:电子工业出版社,2008. 1 – 4.
[17] Ding C, Li T. Adaptive dimension reduction using discriminant analysis and K-means clustering[A]. Ghahramani. Proceedings of the 24th International Conference on Machine Learning [C]. Oregon: ACM, 2007. 521 – 528.
[18] Yin X S, Chen S C, Hu E L. Regularized soft K-means for discriminant analysis[J]. Neurocomputing, 2013, 103: 29 – 42.
[19] Zhu S Z, Wang D D, Li T. Data clustering with size constraints[J]. Knowledge-Based Systems, 2010, 23(8): 883 – 889.
[20] 张腾飞,陈龙,李云. 基于簇内不平衡的粗糙 K-means 聚类算法[J]. 控制与决策,2013,28(10):1479 – 1484.
Zhang T F, Chen L, Li Y. Rough K-means clustering based on unbalanced degree of cluster[J]. Control and Decision, 2013, 28(10): 1479 – 1484. (in Chinese)
[21] Taher N, Babak A. An efficient hybrid approach based on PSO, ACO and K-means for cluster analysis[J]. Applied Soft Computing, 2010, 10(1): 183 – 197.

## 作者简介

**罗　可**　男,1961 年生于湖南长沙,现为长沙理工大学计算机与通信工程学院教授,研究生导师. 主要研究方向为:数据挖掘、计算机应用等研究.
E-mail: luok@csust.edu.cn

**李　莲**　女,1987 年生于湖南郴州,硕士研究生,主要研究方向为:数据库技术、数据挖掘的研究.
E-mail: lilianhappy2012@163.com

**周博翔**　男,1988 年生于湖南邵阳,硕士研究生,主要研究方向为:移动计算、数据挖掘研究.
E-mail: xs_zbx@163.com