Proceedings of the 7th
World Congress on Intelligent Control and Automation
June 25 - 27, 2008, Chongqing, China

A Method of the Knowledge Acquisition Using Rough Set Knowledge Reduction Algorithm Based on PSO

Lin Xu[1,2], Wei Dong[3], Jianhui Wang[1,2], Shusheng Gu[1,2]

[1)] Key Laboratory of Process Industry Automation, Ministry of Education Northeastern University, Shenyang 110004 (E-mail: xulin@ise.neu.edu.cn)

[2)] College of Information Science and Engineering, Northeastern University, Shenyang 110004

[3)] Shenyang Institute of Automation, Chinese Academy of Science, Shenyang 110004

***Abstract*—An variable precision rough set (RS) knowledge acquisition based on discrete particle swarm optimization(DPSO-VPRS) are proposed to solve rough set is lack of the ability of anti-jamming, which is used the information entropy is considered as a suitable function in discrete particle swarm algorithm and the attribute dependent degree of variable precision rough set is optimized, and make the classification rules more reliable in the case of noisy data. The study of knowledge acquisition method based on DPSO-VPRS algorithm which is applied into the grate-kiln system in order to acquire knowledge. The mass production process data is deeply analyzed, and find the key factor which determined the finished pellets quality, then attain manufacturing rule of production process control. The results showed that the grate-kiln expert method is effective and has great value as a reference to the palletizing production process control.**

***Keywords*—particle swarm optimization, variable precision rough set, knowledge reduction, data mining**

一类基于 PSO 的粗糙集知识约简算法的知识获取方法

徐林 [1,2]　董威 [3]　王建辉 [1,2]　顾树生 [1,2]

[1)] 教育部东北大学流程工业自动化重点实验室 沈阳 110004

[2)] 东北大学信息科学与工程学院 沈阳 110004

[3)] 中国科学院沈阳自动化研究所 沈阳 110016

摘　要　针对标准粗糙集（RS）抗干扰能力差的问题，提出了基于离散粒子群的变精度粗糙集知识获取算法（DPSO-VPRS），将信息熵作为适应值函数，对变精度粗糙集的属性依赖度进行寻优，从而在处理噪声数据时获得更可靠的分类规则。研究了基于DPSO-VPRS 算法的知识获取方法，并将其应用到球团矿链篦机－回转窑系统中进行知识获取，通过对大量的生产过程数据进行分析，找出了生产过程中影响球团质量的关键性因素，并得出了生产过程控制的产生式规则。测试结果表明该方法是有效的，对于提高成品球团质量具有较高的参考价值。

关键词　粒子群算法，变精度粗糙集，知识约简，知识获取

1．引言

粗糙集理论(Rough Set，RS)是一种数据推理方法，已被应用于属性约简和规则提取[1,2]。目前，求解知识系统的最小约简是一个 NP-hard 问题，还没有一个最佳的知识约简方法。现有的属性约简算法是根据决策属性对条件属性的依赖性，从核开始，逐次剔除掉那些对分类贡献小的属性，但是此算法并非对所有的知识系统都适用[3]。为了增强粗糙集模型的抗干扰能力，Ziarko 提出了变精度粗糙集模型(Variable Precision Rough Set，VPRS)，该模型引入了阈值的概念，从而允许一定程度的错误分类率存在[4]。

粒子群优化算法(Particle Swarm Optimization, PSO)是一种基于群智能的演化计算技术，已经被广泛应用于函数

国家自然科学基金项目（资助号：60474040）

优化、神经网络训练、模式识别等领域。离散粒子群算法(Discrete PSO，DPSO)是其主要研究方向之一[5-7]。

数据挖掘是在数据库基础上实现的知识发现系统，它利用多种学习手段和方法，从大量的数据中提炼出抽象的知识，从而揭示出蕴涵在这些数据背后的客观世界的内在联系和本质规律。知识获取是数据挖掘的核心。

本文针对标准 RS 抗干扰能力差的问题，引入信息熵为适应值函数，提出了基于离散粒子群的变精度粗糙集知识获取算法（DPSO-VPRS），对基于 DPSO-VPRS 算法的知识获取方法进行了研究，并将其应用到球团矿链篦机—回转窑中进行知识获取，通过对大量的生产过程数据进行分析，找出了生产过程中影响球团质量的关键性因素，并得出了生产过程控制的产生式规则。

2．变精度粗糙集合

2.1 变精度粗糙集中的近似集合

定义 1 论域和知识

设 $S=(U,A,V_a,f_a)$ 是一个知识表达系统，其中 $U\neq\varnothing$ 称为论域，A 表示属性集，V_a 为属性的值域。信息系统用二维信息表示属性的取值，即 f_a 的值。

定义 2 不可分辨关系 $Ind(R)$

设对任给属性集 $R\subset A$，对象 $x_i,x_j\in U$，对 $\forall r\in R$，如果满足 $r(x_i)=r(x_j)$，则称对象 x_i,x_j 对于属性集 R 不可区分。x_i 在 $Ind(R)$ 中的等价类记为 $[x_i]_{Ind(R)}$。

定义 3 粗糙集的 β 上近似和 β 下近似

对于某一 $X\subseteq U$ 和 U 上的一个不可区分关系 $R\subseteq A$，给定阈值 $0.5<\beta\leq 1$，则称

$$\underline{R}_\beta X=\bigcup\left\{[x]_R\left|\frac{|[x]_R\cap X|}{|[x]_R|}\geq\beta\right.\right\} \tag{1}$$

为 X 的 R 下 β 近似集，也称为 X 的 β 正区域，记为 $posr_\beta(X)$。同理

$$\overline{R}_\beta X=\bigcup\{[x]_R \mid \frac{|[x]_R\cap X|}{|[x]_R|}>1-\beta\} \tag{2}$$

为 X 的 R 上 β 近似集，记为 $negr_\beta(X)$。

X 的边界域为

$$bnr_\beta X=\bigcup\{[x]_R \mid 1\text{-}\beta<\frac{|[x]_R\cap X|}{|[x]_R|}<\beta\} \tag{3}$$

X 的 β 负区域为

$$negr_\beta X=\bigcup\{[x]_R \mid \frac{|[x]_R\cap X|}{|[x]_R|}\leq 1-\beta\} \tag{4}$$

将近似集合对 $<\underline{R}_\beta X,\overline{R}_\beta X>$ 称作 X 的变精度粗糙集合。VPRS 模型在标准 RS 中引入参数 β，当 $\beta=1$ 时，VPRS 模型就是标准 RS 模型。当 $\beta<1$ 时，与标准粗糙集的正、负域和边界域相比，VPRS 的正域扩大、负域扩大，从而使边界域变薄，因而对数据的不一致性有一定的容忍度[8]。

定义 4 知识约简（β 近似约简）

在 VPRS 模型中，知识约简是指选择和决策属性 Q 的 β 依赖性相同的最小条件属性子集，且最小条件属性子集不改变原系统的相容性。

设 $red(P,Q,\beta)$ 是条件属性 P 对于 Q 的 β 约简，则有 $red(P,Q,\beta)\subseteq P$，且满足

（1）$\gamma(P,Q,\beta)=\gamma[red(P,Q,\beta),Q,\beta]$；

（2）从 $red(P,Q,\beta)$ 中去掉任何一个属性，都将使(1)不成立。

2.2 决策规则测度分析

对于一个决策规则 $X_i\rightarrow Y_i$ 而言，可以用正确度和覆盖度两个指标来评价其优劣程度。其定义式分别为

$$\alpha_{i,j}=\frac{|x_i\cap y_i|}{|x_i|} \tag{5}$$

$$\lambda_{i,j}=\frac{|x_i\cap y_i|}{|y_i|} \tag{6}$$

规则的正确度反映了当规则的前件成立时规则后件成立的可能性。当正确度小于 1 时，说明在满足规则的条件时，有多个可能的结论，这反映了规则的不一致性。

规则的覆盖度是同时满足规则前件和后件的数据对象在满足规则后件的数据对象中所占的比重。若覆盖度过小规则对数据的代表性不够，从而表现出一定的随机性，这种随机性过大，则其对新数据对象的分类预测能力会大大下降。

定义 5 规则的不确定量度 H^{VPRS}

给定论域 U 及 $C\cup D=A$，$Q\subseteq C$，$U/Q=\{X_1,...,X_S\}$，$U/d=\{\gamma_1,...\gamma_t\}$ 且有 $0\leq\beta<0.5$，标准粗糙集的正域 $V_0=X_1\cup...\cup X_c$，变精度粗糙集的正域 b，则基于信息熵的可变精度粗糙集的不确定量度为

$$\begin{aligned}H^{VPRS}&=H_1+H_2+H_3=\sum_{i\leq c}\frac{|X_i|}{|U|}\lg\frac{|U|}{|X_i|}\\&+\sum_{c<i\leq c+b}\frac{|X_i|}{|U|}\lg\frac{|U|}{|X_i|}+\sum_{i\leq c}\frac{|U\setminus V_1|}{|U|}\lg|U|\end{aligned} \tag{7}$$

2.3 阈值 β 对变精度粗糙规则集的影响

研究阈值 β 的变化对所产生规则集影响的意义在于它能够帮助决策者充分利用规则挖掘算法发现实际问题中的

关键环节，从而起到辅助决策的作用。

如果集合 X 的 β 边界域 $bnr_\beta(X)=\phi$ 或者 $R_\beta X=\overline{R}_\beta X$ 时，称集合 X 为 β 可辨别的，否则称 X 为 β 不可辨别的。集合的可辨别性依赖于 β 的取值。

定理 1 若 X 在阈值 $0.5<\beta\le 1$ 水平上是可辨别的，那么 X 在任何 $\beta_1<\beta$ 上也是可辨别的。

定理 2 若 X 在阈值 $0.5<\beta\le 1$ 水平上是不可辨别的，那么 X 在任何 $\beta_2>\beta$ 上也是不可辨别的。

定义 6 如果集合 X 对每个 β 都是不可辨别的，则称集合 X 为绝对不可辨别的或绝对粗糙的；否则，称 X 为相对粗糙的。

定理 3 集合 X 是绝对粗糙集的充要条件是

$$bnr_{0.5}(X)=\bigcup\left\{[x]_R\,\middle|\,\frac{|[x]_R\cap X|}{|[x]_R|}=0.5\right\}\neq\phi \tag{8}$$

对每一个相对粗糙集 X，都存在一个阈值 β 使得集合 X 在这个阈值水平上是可辨别的。

令 $ndis(R,X)=\{0.5<\beta\le 1\,|\,bnr_\beta\neq\phi\}$，$ndis(R,X)$ 是满足 X 是不可辨别的所有阈值的全体。满足 X 为可辨别的阈值 β 的最大值称为可辨别的阈值，根据定理 1 和定理 2 可知，这个阈值等于 $ndis(R,X)$ 的最大下界，即

$$\xi(R,X)=\min(m_1,m_2) \tag{9}$$

其中：$m_1=1-\max\left\{\frac{|[x]_R\cap X|}{|[x]_R|}\,\middle|\,[x]_R\in U/R,\frac{|[x]_R\cap X|}{|[x]_R|}<0.5\right\}$，

$m_2=\min\left\{\frac{|[x]_R\cap X|}{|[x]_R|}\,\middle|\,[x]_R\in U/R=\frac{|[X]_R\cap X|}{|[X]_R|}\right\}$。

对于变精度粗糙规则集 $Q\to d$，如果存在一个充分小的阈值 β 使得 $V_1=U$，即标准粗糙集意义下的所有不一致规则在 β 水平上均转变成一致性规则，这时称此变精度粗糙规则集为相对粗糙规则集. 相对粗糙规则集在某一 β 水平上完全一致性规则组成，这时我们也称其达到弱完全一致。反过来，如果对一切 β 在变精度粗糙规则集中都含有不一致的规则，则将此规则集称为绝对粗糙规则集。

定理 4 设 U 为一论域，$Q\subseteq C$ 是一条件属性子集 $U/Q=\{X_1,\cdots,X_s\},U/d=\{Y_1,\cdots,Y_t\}$，$0.5<\beta\le 1$，如果存在某一决策类 $Y_j(1\le j\le t)$ 对不可区分关系 Q 是绝对粗糙集，则粗糙规则集 $Q\to d$ 是相对粗糙规则集。

如果 $Q\to d$ 是相对粗糙规则集，则取 $\xi=\min\{\xi(Q,Y_1),\xi(Q,Y_t)\}$，可使相对粗糙规则集 $Q\to d$ 在 $\beta\le\xi$ 时达到弱完全一致。

一般而言，对于确定的 Q,d 和 β，可产生确定的粗糙规则集。当阈值 β 在一定的变化区间内变化时，如果已经产生的粗糙规则集不发生改变，则 β 的这一变化区间为阈值 β 的稳定区间。

3．基于 DPSO 的变精度粗糙集知识约简算法

PSO 算法初始化为一群随机粒子(随机解)。然后通过迭代找到最优解。在每一次迭代中，粒子通过跟踪两个“极值”来进行自我调整。第一个就是粒子本身所经历过的最优解，称为个体极值 p_{best}； 另一个极值是整个种群目前找到的最优解，称为全局极值 g_{best}。

与 PSO 算法不同的是，离散粒子群算法（DPSO）直接将离散值作为待优化变量进行寻优操作，粒子以某一概率、不确定是 1 状态还是 0 状态实现在 1 值和 0 值之间的转换。

对于在 D 维空间中的一个特定的优化问题，DPSO 将任一个体视为 D 维搜索空间的粒子，把染色体编成长度为 n 的二进制位串，每一位对应一个条件属性。

DPSO 初始化产生一个随机矩阵，第一个粒子表示成 $x_i=(x_{i1},x_{i2},\cdots,x_{id})$，在第 i 个例子所有先前历代解中之最佳解的粒子表示为 $p_i=(p_{i1},p_{i2},\cdots,p_{id})$。以指标 g_{best} 表示最佳解的粒子。第 i 个粒子位置变化率表示为 $v_i=(v_{i1},v_{i2},\cdots,v_{id})$。

在二进制的空间中，粒子的移动速度被定义为一个位元从一状态到另一状态变化的机率。因此，一粒子内的任一因次的移动被限制在 0 和 1 的状态空间，其中，任一 v_{id} 代表位元 x_{id} 等于 1 的机率。根据这个定义，p_{id} 和 x_{id} 定义在整数 $\{0,1\}$ 中，而 v_{id} 是个机率值，必须限制在 $[0.0,1.0]$ 的区间内。

v_{id},x_{id} 更新公式如下：

$$\begin{aligned}v_{id,k+1}=&wv_{id,k}+c_1r_1\left(p_{id,k}-x_{id,k}\right)+\\&c_2r_2\left(p_{gd,k}-x_{gd,k}\right)\end{aligned} \tag{10}$$

$$x_{id}=\begin{cases}1, & \text{if } r_1,r_2<S(v_{id})\\0, & \text{else}\end{cases} \tag{11}$$

其中：p_i 为第 i 个粒子先前所有历代解中的最佳解的位置；p_g 为所有粒子中有最佳解的粒子；c_1,c_2 为加速度常数；w 为惯性权重；k 为迭代次数。

粒子在第 k 次迭代中同时向两个极值点接近，个体最优解 $p_{i,k}$（p_{best}）和全局最优解 $p_{g,k}$（g_{best}）。式中下标 i 代表粒子；p_{id} 和 p_{gd} 分别为第 i 个个体最优解和全局最优解的第 d 维分量；x_{id} 和 x_{gd} 分别为对应个体位置矢量的第 d 维分量，$d\in[1,n]$；r_1 和 r_2 为由两个相互独立的随机函数产生的 $[0,1]$ 内的随机变量。

在使用 DPSO 算法进行变精度粗糙集规则获取时，选

用信息熵作为适应值函数进行属性约简

$$F(x)=\lg U-H^{VPRS} \tag{12}$$

其中，$\lg U$ 是 H^{VPRS} 理论最大值。

4．基于 DPSO-VPRS 算法的知识获取方法

数据挖掘是在数据库基础上实现的知识发现系统，它从大量的数据中提炼出抽象的知识，来揭示蕴涵在这些数据背后的客观世界的内在联系和本质规律，实现知识的自动获取。

基于 DPSO-VPRS 算法的知识获取由数据准备（包括数据清洗、数据选择、数据预处理、数据表示）、对象分类、对象重要性分析、属性之间的依赖关系分析、数据约简和求核（属性核、值核）、决策算法、规则生成、规则合并、知识表示、评价等部分组成。

4.1 数据的预处理

数据的预处理主要是对目标数据库中的数据进行再加工，检查数据的完整性以及数据的一致性，对其中的噪声数据的处理，包括数据清理、数据集成与变换、数据的离散化等。

4.2 条件属性集和决策属性集的选择

在经过数据预处理后，用户要作出条件属性集和决策属性集的选择。对于形如 $X\Rightarrow Y$ 的规则，X 是规则的前件，Y 是规则的后件，这是待挖掘规则的框架。

选择的过程：用户将关系表中所有的属性分成两大类，一类是条件属性集 C，另一类是决策属性集 D，使得 $C\cap D=\Phi$ 且 $C\cup D=A$（A 为关系表中所有属性的集合）。一般情况下，条件属性集包含多个属性，决策属性集只包含一个决策属性。

4.3 条件属性集的约简

由于选择的条件属性集并没有经过严格的数学验证，其中可能存在一些冗余的属性，为提高系统的速度和性能以及最终产生的规则的数量和质量，必须对条件属性集进行约简，删除掉一些冗余的属性。

这里使用 DPSO-VPRS 知识约简算法进行约简。

属性近似约简步骤：

输入：原始数据，设定 $0.5<\beta\le 1$，群体规模 m。

输出：属性子集。

（1）由随机产生的 m 个长度为 n 的二进制串所代表的个体组成初始种群；由 DPSO 算法可知，需要构造适合用离散解值集和离散速度值集表示的数据表的各个属性值。

（2）对每个个体 x，由式(7)计算出所含条件属性在变精度粗糙集规则意义下的信息熵量度 $H^{VRPS}(x)$，并由式(12)算出每个个体的适应值 $F(x)$。

（3）使用式（10）计算 v_i，使用式（11）更新 x_i。

（4）计算每个粒子的新位置的适应值；对每个粒子，若粒子的适应值优于原来的个体极值 p_{best}，设置当前适应值为个体极值 p_{best}；根据各个粒子的个体极值 p_{best} 找出全局极值 g_{best}。

（5）如果停止条件满足，g_{best} 为最后约简，否则回到步骤(2)。

经过属性约简的决策表可以看做一个决策规则集合。每条决策规则都精确地对应约简后的决策表中的一个实例。采用下述的值约简算法来化简每个实例对应的决策规则：

（1）对于一个实例，去掉一个条件属性，然后检查它与表中其他实例是否产生新的矛盾。

（2）如果产生了新的矛盾，则恢复去掉的条件属性；否则转到(3)。

（3）对该实例的所有条件属性进行与(1)，(2)相同的操作，便得到该实例对应的最大广义决策规则。

对所有实例进行与(1)，(2)，(3)相同的操作，便得到决策规则集。

当条件属性集 C 有多个约简时，可以比较决策属性集 D 与每一个约简的依赖程度，从而给出依赖程度最小的约简。依赖度的评价标准可用“条件熵”来判断。

4.3 规则的生成与综合

经过属性约简后，得到的是一个由约简了的条件属性集和决策属性集构成的新表，这个表也可称为一个“决策表”，但还不能就将其中的每一条记录作为最终的规则。决策算法中的每一条规还要进行约简，以删去规则中冗余的属性。

5．应用研究

球团矿链篦机－回转窑法的工艺流程包括精矿配料、精矿干燥、辊压、膨润土与灰尘配料、混合、造球、生球布料、生球干燥、预热、氧化焙烧、冷却及成品输出等。图 1 给出了球团矿链篦机－回转窑控制系统结构。

下面说明链篦机－回转窑知识获取的整个过程。

5.1 数据的预处理

5.1.1 属性的选择

根据球团生产的热工制度，影响成品球团质量的主要过程参数有料层的厚度 a，链篦机的速度 b，回转窑的速度 c，环冷机的速度 d，鼓风干燥段的温度 e，抽风干燥段

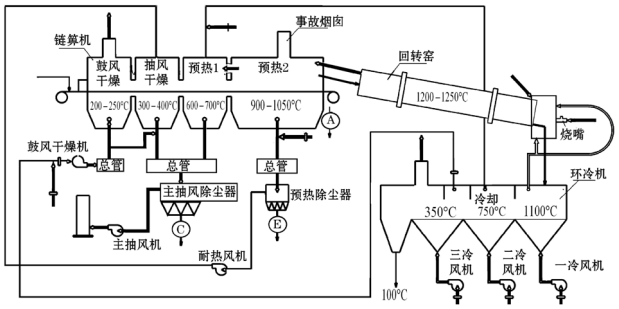

图 1 链篦机－回转窑结构图

的温度 f，预热 I 段的温度 g，预热 II 段的温度 h，回转窑窑头的温度 i，回转窑窑尾的温度 j，环冷机三段的温度 k，所以选择上述属性为条件属性集 C，即 $C=\{a,b,c,d,e,f,g,h,i,j,k\}$，而决策属性集 D 即为成品球团质量 I，即 $D=\{I\}$，则属性全集为 $A=\{a,b,c,d,e,f,g,h,i,j,k,I\}$。

5.1.2 属性的离散化

传统的离散化方法如等宽区间法等，操作简单，使用方便，但是往往容易造成数据分布不均，丢失部分信息。针对这种情况，根据系统所有条件属性均为连续属性，采用文献[6]所提出的连续属性离散化方法，利用决策属性依赖度作为反馈信息，在保持原始决策属性依赖度不变的前提下，寻找使得约简效率最高的划分。该算法能在保证决策表原始分类能力不变的前提下，提高约简效率同时，各个属性拥有较少的分割区间，使规则集合更加简洁。

利用上述方法对采集的链篦机－回转窑的生产过程数据进行离散化，可以得到如表 1 所示的条件属性离散化参数表。

5.1.3 初始决策表的形成

根据选择的属性以及离散化处理后的属性值，首先在数据库中删除重复的实例，从而建立起初始的决策表，如表 2 所示(这里仅列出其中的一部分)。

5.2 决策表的一致性分析

设链篦机－回转窑的知识表达系统为 $S=\{U,A\}$，其中 $U=\{1,2,3\cdots,111,112\}$，为所讨论的个体集，即论域，$A=\{a,b,c,d,e,f,g,h,i,j,k,I\}$ 为属性集合，包含条件属性集 $C=\{a,b,c,d,e,f,g,h,i,j,k\}$ 和决策属性集 $D=\{I\}$，有 $A=C\bigcup D$ 且 $C\bigcap D=\Phi$。

下面分析一下初始决策表 2 的一致性。首先求出论域 U 由各个条件属性（等价关系）的分类情况，以条件属性 a 为例：

$U|Ind(a)$ ={{1, 2, 3, 4, 5, 6, 9, 10, 13, 14, 16, 17, 18, 20, 22, 24, 25, 26, 29, 30, 32, 33, 34, 35, 36, 37, 40, 41, 42, 43, 46, 47, 50, 51, 52, 54, 55, 62, 64, 65, 66, 67, 68, 70, 71, 72, 74, 75, 79, 81, 83, 88, 89, 91, 92, 94, 95, 96, 103, 105, 107, 108, 109, 112}, {7, 8, 11, 12, 15, 19, 21, 23, 27, 28, 31, 38, 39, 44, 45, 48, 49, 53, 56, 57, 58, 59, 60, 61, 63, 69, 73, 76, 77, 78, 80, 82, 84, 85, 86, 87, 90, 93, 97, 98, 99, 100, 101, 102, 104, 106, 110, 111}}

同理，可以求出 $U|Ind(a,b)$、$U|Ind(a,b,c)$、…、$U|Ind(a,b,c,d,e,f,g,h,i,j,k)$。

由此可以求出决策属性集 D 的 C 正区域为 $posr_{\beta}(D)$={1, 2, 3, 4, …, 108, 109, 110, 112}。

决策属性集 D 对于条件属性集 C 的依赖程度：

$$k=\frac{card\left(posr_{\beta}(D)\right)}{card(U)}=\frac{112}{112}=1 \tag{13}$$

由此可见决策属性集 D 完全依赖于条件属性集 C，此初始决策表为一致性决策表。

表 1 条件属性离散化参数表

属性	1	2	3	4	5	6
a	[170,189.5)	[189.5,210]				
b	[3.1,3.2)	[3.2,3.3]				
c	[1,1.4]					
d	[1,1.3]					
e	[42,65)	[65,88]				
f	[278,297)	[297,316)	[316,335]			
g	[600,702]					
h	[976,1012)	[1012,1048]				
i	[1040,1067)	[1067,1093.3)	[1093.3,1120)	[1120,1146.7)	[1146.7,1173.3)	[1173.3,1200]
j	[900,988]					
k	[397,408.6)	[408.6,420.2)	[420.2,431.8)	[431.8,443.4)	[443.4,455]	

表 2 初始决策表

U	*a*	*b*	*c*	*d*	*e*	*F*	*g*	*h*	*i*	*J*	*k*	*L*
1	1	2	1	1	2	3	1	2	6	1	4	1
2	1	2	1	1	1	2	1	2	5	1	4	1
3	2	2	1	1	2	2	1	2	6	1	5	1
4	1	1	1	1	1	3	1	2	6	1	3	1
5	1	1	1	1	1	3	1	1	3	1	3	2
6	1	2	1	1	1	3	1	2	5	1	2	1
7	2	2	1	1	1	2	1	2	6	1	5	1
8	2	2	1	1	1	2	1	2	6	1	3	1
9	2	1	1	1	1	1	1	2	3	1	5	2
10	2	2	1	1	1	1	1	2	5	1	3	1
…												

5.3 决策表的约简

取 $m=20$, $\alpha=16$, $c_1=c_2=2$ ， $\beta=0.9$ ， 最大迭代次数为 $G=200$ 。计算可得条件属性对决策属性的最小近似约简规则表示为 $\beta=0.7$ 时，该决策规则达到弱完全一致，β 的阈值稳定区间为[0.73,0.98]，约简后的决策表见表 3。

$a(1)\wedge b(2)\wedge e(2)\wedge h(2)\wedge i(4)\rightarrow l(1)$,**正确率**=1.0,**覆盖度**=0.86

$a(1)\wedge b(2)\wedge e(1)\wedge h(2)\wedge i(4)\rightarrow l(1)$,**正确率**=0.95,**覆盖度**=0.63

$a(1)\wedge b(1)\wedge e(1)\wedge h(1)\wedge i(1)\rightarrow l(2)$,**正确率**=0.89,**覆盖度**=0.66

$a(2)\wedge b(1)\wedge e(2)\wedge h(1)\wedge i(1)\rightarrow l(2)$,**正确率**=0.97,**覆盖度**=0.46

……

表 3 简化的决策表

U	*a*	*b*	*e*	*h*	*i*	*L*
1	1	2	2	2	4	1
2	1	2	1	2	4	1
3	1	1	1	1	1	2
4	2	1	2	1	1	2
5	2	1	2	2	2	2
6	1	1	2	1	1	2
7	2	2	1	2	4	1
8	2	2	2	2	4	1
9	1	1	1	2	1	2
10	1	2	2	2	5	1
…						

5.4 决策规则的生成与综合

由最终的约简表可以得到一系列的产生式规则，列举其中的两条如下，其中符号 $a(1)$ 表示属性 a 的离散值为 1，其余类似：

$$a(1)\wedge b(2)\wedge e(2)\wedge h(2)\wedge i(4)\rightarrow l(1)$$

$$a(1)\wedge b(1)\wedge e(1)\wedge h(1)\wedge i(1)\rightarrow l(2)$$

上述符号式表示，当链篦机－回转窑各段的温度都在规定的范围内时，有如下参考规则：

规则 1 当料层厚度范围在[170, 189.5)之间，链篦机机速范围在[3.2, 3.3]之间，鼓风干燥段温度范围在[65, 88]之间，预热Ⅱ段温度范围在[1012, 1048]之间，回转窑窑头的温度范围在[1120, 1146.7)之间时，生产的球团质量为一级品；

规则 2 当料层厚度范围在[170, 189.5)之间，链篦机机速范围在[3.1, 3.2)之间，鼓风干燥段温度范围在[42, 65)之间，预热Ⅱ段温度范围在[976, 1012)之间，回转窑窑头的温度范围在[1040, 1067)之间时，生产的球团质量为二级品。

由此可见，经过对初始决策表的约简，其复杂度大大降低，非常有利于规则存储与检索，而且得到的规则具有较高的正确度和覆盖度。

6．结论

本文提出了基于离散粒子群的变精度粗糙集知识获取算法（DPSO-VPRS），引入信息熵为适应值函数，对变精度粗糙集的属性依赖度进行寻优，从而在处理噪声数据时获得更可靠的分类规则，研究了基于 DPSO-VPRS 算法的知识获取方法，并将其应用到球团矿链篦机－回转窑知识获取中。测试结果表明该方法简单有效，不但能够消除噪声的影响，而且得到的规则具有较高的正确度和覆盖度，提高了数据分析的鲁棒性。

参考文献

[1] 王莉,孙一康. 基于递阶遗传算法的 RBF 神经元网络板形板厚综合控制. *计算机仿真*, vol.20, no.2, pp.82-85, 2003.

[2] J. Kennedy and R. Eberhart, “Particle Swarm Optimization,” in *Proc. of the 1995 IEEE International Conference on Neural Networks*, Perth, Australia, 1995, pp. 1942-1948.

[3] 孙一康. 带钢热连轧的模型与控制. 北京: 冶金工业出版社, 2002.

[4] Y. Shi and R. Eberhart, “A Modified Particle Swarm Optimizer,” in *IEEE International Conference of Evolutionary Computation*. Anchorage, Alaska, 1998, pp. 69-73.

[5] 吕振肃, 侯志荣. “自适应变异的粒子群优化算法,” *电子学报*, vol.32, no.3, pp.416-420, 2004.

[6] 骆晨钟, 邵惠鹤. “采用混沌变异的进化算法,” *控制与决策*, vol.15, no.5, pp.557-560, 2000.

[7] J. Moody and C. Darken, “Fast learning in networks of locallytuned processing units,” *Neural Computation*, no.1, pp. 281-294, 1989.