

## 一种决策表属性值细化的正域约简算法

唐定勇<sup>1</sup>, 景运革<sup>2</sup>

(1 西南交通大学 CAD 工程中心, 四川 成都 610031;

2 西南交通大学 信息科学与技术学院, 四川 成都 610031)

**摘 要:** 目前决策表的数据信息都是动态变化的, 为了有效地对这些数据集进行处理, 提出了信息系统属性值细化时粗糙集增量式约简的矩阵方法并构造相应的算法, 最后, 通过实例分析来说明属性约简的具体操作方法和算法的有效性与可行性。

**关键词:** 粗糙集; 增量式更新; 正域约简; 关系矩阵; 属性值细化

**中图分类号:** TP 311

**文献标识码:** A

**文章编号:** 1000-7180(2015)03-0023-05

A Reduction Algorithm of Positive Domain for Decision Table  
Based on Values RefiningTANG Ding-yong<sup>1</sup>, JIN Yun-ge<sup>2</sup>

(1 Engineering Center of CAD, Southwest Jiaotong University, Chengdu 610031, China;

2 School of Information Science and Technology, Southwest Jiaotong University, Chengdu 610031, China)

**Abstract:** Many real data increase dynamically in size. As an effective and efficient mechanism to deal with such data, Incremental algorithms for attribute reduction based on modified equivalence relation matrix are proposed, by which minimal attribute reduction of new decision table can be obtained quickly when attribute values are refined. Finally, Example confirms the feasibility and effectiveness of proposed operation and method of attribute reduction.

**Key words:** Rough set; Incremental update; Positive domain reduction; Relation matrix; Values refining

## 1 引言

现有的属性约简方法大多是针对静态的信息系统, 如何在属性值细化的条件下快速高效地找到决策表的属性约简, 已成为基于粗糙集的动态知识发现领域中的一个研究热点。当面临数据不断变化的动态环境时, 用传统的知识获取方法对复杂数据进行挖掘, 很难及时、高效地从动态数据中提取出有用的信息及知识。增量式更新方法, 通过更新部分变化的数据, 能提高操作效率, 满足数据挖掘需求。文献[1]中提出了一种利用粗糙集技术的增量式特征选择的方法; 文献[2]中提出了正向近似和逆向近似, 并成功应用于启发式约简算法的加速, 为基于粗糙

集的知识发现性能优化提供了新思路; 刘薇等分析了在对象动态增加情况下信息熵的变化机制, 提出了一种基于条件熵的快速增量约简方法<sup>[3]</sup>; 提出了一种利用属性值动态变化的属性约简方法<sup>[4]</sup>; 王磊等提出了信息系统的属性值发生变化时变精度粗糙集模型中概念上、下近似集增量式更新的矩阵方法并构造出相应的算法<sup>[5]</sup>。在优势关系粗糙集模型中, 给出数据库中属性值粗化细化的概念, 提出了属性值发生粗化细化时近似集的增量更新算法。但是当属性值发生细化, 通过增量式更新等价关系矩阵, 并通过等价关系矩阵相乘的约简方法却鲜有报道。

矩阵理论是处理大量复杂数据的有效工具, 本文在信息系统的属性值发生细化时, 运用矩阵作为

收稿日期: 2014-05-03; 修回日期: 2014-07-01

基金项目: 国家自然科学基金联合项目(U1230117); 国家自然科学基金项目(61175047)

表达和计算工具,利用既有的约简知识,通过增量式学习方法,能够在较短的时间内得到一个新的属性约简.

## 2 粗糙集基础知识

### 2.1 基本概念

**定义 1** 信息系统是一个四元组  $S = (U, A = C \cup D, V, f)$ ,  $U = \{x_1, x_2, \dots, x_n\}$  为论域;  $C, D$  分别表示条件属性集和决策属性集,  $V = \bigcup_{a \in C \cup D} V_a$ , 其中  $V_a$  是属性  $a$  的值域,  $f: U \times C \cup D \rightarrow V$  是一个信息函数, 即  $a \in C \cup D, x \in U$  有  $f(x, a) \in V_a$  [6].

**定义 2** 设知识库  $K = (U, R), P, Q \subseteq R$ ,  $POS_P^Q = \bigcup R$ , 知识  $Q$  依赖于  $P$  定义为  $\gamma_P(Q) = \frac{|POS_P^Q|}{|U|}$ .

**定义 3** 矩阵乘法: 如果  $A = (a_{ij})_{m \times n}, B = (b_{ij})_{n \times p}$ , 则  $A \cdot B = (c_{ij})_{m \times p}$ , 其中  $(c_{ij})_{m \times p} = \sum_{k=1}^n a_{ik} \cdot b_{kj}$ .

**定义 4** 信息系统  $S = (U, A = C \cup D, V, f)$ ,  $T_C$  是论域  $U$  上的等价关系, 则其等价关系矩阵  $M_{n \times n}^{T_C} = (m_{i,j})_{n \times n}$  的元素为 [7]

$$(m_{i,j})_{n \times n} = \begin{cases} 1, & (u_i, u_j) \in T_C \\ 0, & (u_i, u_j) \notin T_C \end{cases}$$

**定义 5** 信息系统  $S = (U, A = C \cup D, V, f)$ , 决策属性  $D$  的等价类  $U/D = \{D_1, D_2, \dots, D_m\}, D_i$  是  $U/D$  的任意子集. 则决策属性的矩阵  $M_{n \times m}^{D}$  表示:

$$M_{n \times m}^D = (m_{i,j})_{n \times m} = \begin{cases} 1, & u_i \in D_j \\ 0, & u_i \notin D_j \end{cases}$$

### 2.2 非增量式矩阵约简方法

针对静态信息系统, 文献 [7] 中给出了利用决策条件属性的等价关系矩阵和诱导矩阵来计算决策属性的正域, 并提出了矩阵的属性约简算法 1.

决策表  $S$  中的等价关系为  $T_C$ , 其等价关系矩阵  $M_{n \times n}^{T_C} = (m_{i,j})_{n \times n}$ , 矩阵  $M_{n \times n}^{T_C}$  的诱导矩阵  $\Lambda_{n \times n}^{T_C}$  表示为  $\Lambda_{n \times n}^{T_C} = \text{diag}(1/\sum_{k=1}^n m_{1k}, 1/\sum_{k=1}^n m_{2k}, \dots, 1/\sum_{k=1}^n m_{nk})$

**定理 1** 信息系统  $S = (U, A = C \cup D, V, f)$ , 令  $U/C$  和  $U/D$  为  $U$  中的两个划分, 其中  $U/D = \{D_1, D_2, \dots, D_m\}, D_i$  是  $U/D$  的任意子集, 决策属性  $D$  关于条件属性  $C$  的正域矩阵表示:  $POS_C(D) = (\Lambda_{n \times n}^{T_C} \cdot (M_{n \times n}^{T_C} \cdot M_{n \times m}^D))_1 = (\Lambda_{n \times n}^{T_C} \cdot C_{n \times m})_1$  且

$(\Lambda_{n \times n}^{T_C} \cdot (M_{n \times n}^{T_C} \cdot M_{n \times m}^D))_1$  是矩阵  $\Lambda_{n \times n}^{T_C} \cdot (M_{n \times n}^{T_C} \cdot M_{n \times m}^D)$  的 1-截矩阵.

**定理 2** 设决策信息系统  $S = (U, A = C \cup D, V, f), C = \{a_1, a_2, \dots, a_n\}$  是条件属性集,  $B \subseteq C$  且  $B \neq \emptyset, a_i \in C - B$ , 属性  $B, a_i$  的等价关系矩阵分别为  $M_{n \times n}^B = (m_{ij})_{n \times n}, M_{n \times n}^{a_i} = (m_{ij}')_{n \times n}$ , 则把属性  $\{a_i\}$  添加到属性集  $B$  后,  $B$  的等价关系矩阵更新为  $M_{n \times n}^{B \cup \{a_i\}}$  的元素为  $(m_{ij})^+ = (\min(m_{ij}, m_{ij}'))$ .

**定理 3** 设信息系统  $S = (U, A = C \cup D, V, f), B \subseteq C$  且  $B \neq \emptyset, a_i \in A - B, \Lambda_{n \times n}^{T_B} = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$  ( $\lambda_i = \sum_{j=1}^n m_{ij}$ ) [8].

设  $\Lambda_{n \times n}^{T_{B \cup \{a_i\}}} = \text{diag}(\lambda_1^+, \lambda_2^+, \dots, \lambda_n^+)$  ( $\lambda_i^+ = \sum_{j=1}^n m_{ij}$ ),

则有  $\lambda_i^+ = \lambda_i - \sum_{j=1}^n (1 - \omega_{i,j}) \cdot m_{ij}$

### 2.3 决策信息系统属性约简算法的描述

由定理 1 ~ 3 设计了矩阵的属性约简算法 1 [7].

**输入:** 决策信息系统  $S = (U, A = C \cup D, V, f)$ .

**输出:** 决策表的属性约简  $RED$ .

Setp1 计算  $D$  关于  $C$  的正域  $POS_C^D$ ;

Step2 计算  $D$  关于  $C$  的依赖度并降序排列, 假设顺序为  $\gamma_{C_1}(D) > \gamma_{C_2}(D) > \dots > \gamma_{C_m}(D)$ ;

Setp3  $RED = \varnothing$ ;

Setp4 for  $i = 1:m$

4.1 把属性  $\{C_i\}$  添加到  $RED$ , 利用定理 2 计算属性  $RED \cup \{C_i\}$  的等价关系矩阵  $M_{n \times n}^{T_{RED \cup \{C_i\}}}$ .

4.2 把属性  $\{C_i\}$  添加到  $RED$ , 利用定理 3 计算  $RED \cup \{C_i\}$  的对角矩阵为

$$\Lambda_{n \times n}^{RED} = \text{diag}(\lambda_1^+, \lambda_2^+, \dots, \lambda_n^+) (\lambda_i^+ = \sum_{j=1}^m m_{ij})$$

4.3 把属性  $\{C_i\}$  添加到  $RED$ , 利用定理 3 计算  $D$  关于  $RED \cup \{C_i\}$  的正域  $POS_{RED}(D)$

if  $POS_C(D) = POS_{RED}(D)$

Then

$RED = RED \cup C_i$

End

For  $(a' \in RED)$

If  $(POS_{RED - \{a'\}}(D) = POS_{RED}(D))$  then

$RED = RED - a'$

End if

End

Setp5 输出决策表的约简  $RED$ .

### 3 属性值细化的矩阵约简方法

#### 3.1 属性值细化的概念

**定义 6** 信息系统  $S = \langle U, A = C \cup D, V, f \rangle$ ,  $B \subseteq A$  且  $B \neq \varphi, a_l \in B$ , 属性  $a_l$  的值域为  $V_l^{[5]}$ ,  $[x_i]_{a_l} = \{x_i \in U \mid f(x_i, a_l) = f(x_j, a_l)\}$ , 设  $Y \subset [x_i]_{a_l}$  且  $Y \neq \varphi$ , 对于任意  $x_k \in Y$ , 若有  $f(x_k, a_l) = v$ , 则称属性值  $f(x_i, a_l)$  细化为  $v$ .

记  $X = [x_i]_{a_l}, Y = \{x_m \in U \mid f(x_m, a_l) = v\}$ , 则  $X - Y = \{x_n \in U \mid f(x_n, a_l) \neq f(x_i, a_l)\}$  为  $X$  集合中属性  $a_l$  的值在细化前保持保持不变的集合.

记  $I_{X-Y} = \{i \mid u_i \in X - Y\}, I_Y = \{i \mid u_i \in Y\}$ ,  $I_{X-Y}$  表示集合  $X - Y$  中的元素下标组成的集合,  $I_X$  表示  $Y$  中的元素下标组成的集合.

#### 3.2 属性值细化更新机制

假设有一个动态决策表, 当属性值随着时间而发生变化, 重新计算属性等价关系需要花费大量的时间, 为了克服这个缺陷, 增量式学习方法在细化化前的等价关系矩阵  $M_{n \times n}^{T_{\varphi}}$  基础上, 通过快速更新原来的等价关系矩阵而得到细化化后的等价关系矩阵  $M_{n \times n}^{T_{\varphi}^{\uparrow}}$ , 当决策表属性值发生细化时, 定理 4 的机制介绍了等价关系矩阵更新机制, 然后利用定理 1 得到新的正域.

**定理 4** 属性  $a_i$  的属性值发生细化时, 当属性  $a_i$  的值细化后, 其对应的等价关系矩阵  $M_{n \times n}^{T_{\varphi}^{\uparrow}}$  中元素的表示为<sup>[5]</sup>

$$(m_{i,j}^{\uparrow})_{n \times n} = \begin{cases} 0 & i \in I_{X-Y} \text{ 且 } j \in I_Y \\ 0 & i \in I_Y \text{ 且 } j \in I_{X-Y} \quad 1 \leq i, j \leq |U| \\ m_{i,j} & \text{其他} \end{cases}$$

#### 3.3 属性值细化更新机制

根据等价关系矩阵增量式更新机制, 当决策表中属性值发生变化(细化)时, 我们依据定义 1 ~ 5 和定理 1 ~ 4 提出了增量式矩阵约简算法 2, 所提算法是在原有约简的基础上, 通过快速更新等价关系矩阵, 能在较短的时间内找到一个新的属性约简.

**算法 2** 粗糙集增量式矩阵约简算法

**输入:** 细化前等价关系  $M_{n \times n}^{T_{\varphi}}$  和约简的等价关系矩阵  $M_{n \times n}^{T_{\varphi}^{RED}}$  以及属性的约简  $RED$ , 由粗糙集静态

属性约简的矩阵算法得出.

**输出:** 属性值发生细化后的属性约简.

Setp1: 根据定理 4 增量式更新等价关系矩阵  $M_{n \times n}^{T_{\varphi}}$  和  $M_{n \times n}^{T_{\varphi}^{RED}}$ ;

Step2:  $RED^{\uparrow} = RED$ ;

Step3: 根据定理 1 计算更新后的  $POS_C(D)^{\uparrow}$  和  $POS_{RED}(D)^{\uparrow}$ , 如果  $POS_C(D)^{\uparrow} = POS_{RED}(D)^{\uparrow}$  转到 Setp5, 否则执行 Setp4;

Setp4: for  $i = 1: |C - RED|$

4.1 按照属性依赖度的大小来增加属性  $C_i$ ;

4.2 根据定理 2 增量式更新等价关系矩阵  $M_{n \times n}^{T_{\varphi}^{RED}^{\uparrow}}$ ;

4.3 根据定理 3 增量式更新对角矩阵  $\Lambda_{n \times n}^{T_{\varphi}^{RED}^{\uparrow}} = \text{diag}(\lambda_1^+, \lambda_2^+, \dots, \lambda_n^+)$  ( $\lambda_i^+ = \sum_{j=1}^n m_{ij}$ );

4.4 根据定理 1 计算细化后  $D$  关于  $RED^{\uparrow}$  的正域  $POS_{RED}(D)^{\uparrow}$ ;

if  $POS_C(D)^{\uparrow} = POS_{RED}(D)^{\uparrow}$

Then

$RED^{\uparrow} = RED \cup C_i$

End if

End

For ( $a' \in RED^{\uparrow}$ )

If ( $POS_{RED^{\uparrow} - \{a'\}}(D) = POS_{RED^{\uparrow}}(D)$ ) then

$RED^{\uparrow} = RED^{\uparrow} - a'$

End if

End

Setp5: 输出细化后最小属性约简  $RED^{\uparrow}$ .

### 4 算例

**决策表如表 1:** 论域  $U = \{x_1, x_2, x_3, x_4, x_5, x_6\}$ , 条件属性  $C = \{a, b, c\}$ , 决策属性  $D = \{d\}$ . 决策表的属性约简为  $\{a, b\}$ , 如果属性值细化为  $f(x_3, b) = 1$ , 属性值细化后的属性约简.

表 1 决策表

| $U$   | $a$ | $b$ | $c$ | $d$ |
|-------|-----|-----|-----|-----|
| $x_1$ | 2   | 2   | 0   | 1   |
| $x_2$ | 1   | 2   | 0   | 0   |
| $x_3$ | 1   | 2   | 0   | 1   |
| $x_4$ | 0   | 0   | 0   | 0   |
| $x_5$ | 1   | 0   | 1   | 0   |
| $x_6$ | 2   | 0   | 1   | 1   |

#### 4.1 计算决策属性的列矩阵 $M_{n \times m}^D$

根据表 1 和定义 5, 决策属性的列矩阵  $M_{n \times m}^D$  为

$$M_{n \times m}^D = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}$$

#### 4.2 增量式计算等价矩阵 $M_{n \times n}^{T_c}$ 和 $M_{n \times n}^{T_{RED}}$

条件属性  $C$  的等价关系矩阵  $M_{n \times n}^{T_c}$  和  $RED$  的等价关系矩阵  $M_{n \times n}^{T_{RED}}$  可从文献[8] 得, 即:

$$M_{n \times n}^{T_c} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$M_{n \times n}^{T_{RED}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

当属性值细化后, 根据定理 4 用增量式方法计算条件属性  $C$  的等价关系矩阵  $M_{n \times n}^{T_c}$  以及  $RED$  的等价关系矩阵  $M_{n \times n}^{T_{RED}}$ :

$$M_{n \times n}^{T_c} = \begin{bmatrix} 1 & 0 & \mathbf{0} & 0 & 0 & 0 \\ 0 & 1 & \mathbf{0} & 0 & 0 & 0 \\ \mathbf{0} & \mathbf{0} & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$M_{n \times n}^{T_{RED}} = \begin{bmatrix} 1 & 0 & \mathbf{0} & 0 & 0 & 0 \\ 0 & 1 & \mathbf{0} & 0 & 0 & 0 \\ \mathbf{0} & \mathbf{0} & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

#### 4.3 计算属性值细化后的约简

根据定理 3 计算属性值细化后的诱导矩阵

$\Lambda_{n \times n}^{T_c}$  和  $\Lambda_{n \times n}^{T_{RED}}$ :

$$\Lambda_{n \times n}^{T_c} = \text{diag}(1/1, 1/1, 1/1, 1/1, 1/1, 1/1),$$

$$\Lambda_{n \times n}^{T_{RED}} = \text{diag}(1/1, 1/1, 1/1, 1/1, 1/1, 1/1).$$

根据定理 1 计算属性值细化后的条件属性  $C$  和约简  $RED$  正域

$D$  关于  $C$  的正域为

$$POS_C(D)^{\uparrow} = (\Lambda_{n \times n}^{T_c} \cdot (M_{n \times n}^{T_c} \cdot M_{n \times m}^D))_1$$

$$= \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix};$$

$$POS_C(D)^{\uparrow} = \{x_1, x_2, x_3, x_4, x_5, x_6\}.$$

$D$  关于  $RED$  的正域为

$$POS_{RED}(D)^{\uparrow} = (\Lambda_{n \times n}^{T_{RED}} \cdot (M_{n \times n}^{T_{RED}} \cdot M_{n \times m}^D))_1$$

$$= \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix};$$

$$POS_{RED}(D)^{\uparrow} = \{x_1, x_2, x_3, x_4, x_5, x_6\}.$$

故  $POS_C(D)^{\uparrow} = POS_{RED}(D)^{\uparrow}$ .

所以, 表 1 属性  $b$  的值细化后的约简为  $\{a, b\}$ .

## 5 实验测试与分析

当属性值细化后, 为了验证增量式约简的矩阵算法比非增量式约简的矩阵算法的有效性, 我们从 UCI 数据集下载了 2 个数据集分别为 Wine, Balance. 数据集描述见表 2, 分别用两种方法对 2 个数据集进行了测试, 并对所消耗的时间进行比较, 实验测试的软硬件环境: CPU IntelCore™ 双核 2 GHz, 内存 1.0 GB, Windows 7.0 操作系统, C++ 开发平台.

表 2 数据集描述

| 序号 | 数据集     | 属性个数 | 对象个数 |
|----|---------|------|------|
| 1  | Wine    | 13   | 178  |
| 2  | Balance | 5    | 625  |

在测试实验中, 把每个数据集均匀划分为 5 个大小的数据子集, 其作用就是测试属性值细化的对象数不同所消耗时间的变化规律, 比较增量式和非增量式的矩阵算法所消耗的时间, 实验测试结果如图 1 所示, 其中横轴表示数据子集的对象数占总对

象数的百分比,纵轴表示计算约简所消耗的时间。

当属性值细化时,分别用增量式矩阵约简和非增量式矩阵约简方法对上述的两个数据集做测试,测试结果如图1所示,得出结论如下所示。

(1)增量式矩阵约简算法较非增量式矩阵约简算法效率较高,主要表现在增量式约简算法所消耗的时间小于非增量式约简算法。

(2)针对同一数据集,随着数据对象的增加,增量式和非增量式约简算法所消耗的时间均呈增长趋势,只是增量式约简算法所消耗的时间增加较平缓。

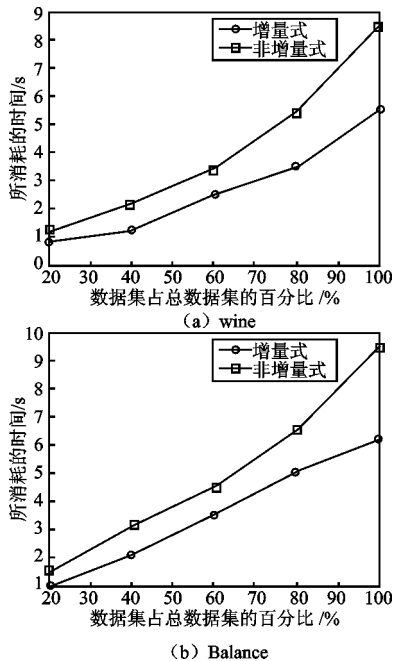


图1 属性值细化时增量式约简和非增量式约简的时间消耗比较

## 6 结束语

当决策表中一些属性值发生细化时,常常会引起属性约简的变化,而重新计算属性约简,将会造成大量重复计算,效率不高.为此给出属性值发生细化条件下矩阵约简的方法.该算法充分利用了更新前决策表的等价关系矩阵和属性约简等信息,能快速找到新的约简,缩短了属性约简的时间,保证了属

性约简的实时性,在理论分析的基础上给出了实例,说明了基于属性值细化的决策表正域约简算法是可行和有效的。

## 参考文献:

- [1] Jiye Liang, Feng Wang, Chuangyin Dang, et al. A group incremental approach to feature selection applying rough set technique[J]. IEEE Transactions on Knowledge and Data Engineering, 2012(9): 1-31.
- [2] Yuhua Qian, Jiye Liang, Witold Pedrycz, et al. Positive approximation: an accelerator for attribute reduction in rough set theory[J]. Artificial Intelligence, 2010, 174(9/10): 597-618.
- [3] 刘薇, 梁吉业, 魏巍, 等. 一种基于条件熵的增量式属性约简算法[J]. 计算机科学, 2011(1): 229-231.
- [4] Feng Wang, Jiye Liang, Chuangyin Dang. Attribute reduction for dynamic data sets[J]. Applied Soft Computing, 2012(18): 1-18.
- [5] 王磊, 洪志全, 万旋. 属性值变化时变精度粗糙集模型中近似集动态更新的矩阵方法研究[J]. 计算机应用研究, 2013, 30(7): 2011-2013.
- [6] 刘清. Rough set 及 Rough 理[M]. 北京: 科学出版社, 2001: 7-16.
- [7] Junbo Zhang, Tianrui Li, Da Ruan, et al. Rough sets based matrix approaches with dynamic attribute variation in set-valued information systems[J]. International Journal of Approximate Reasoning, 2012, 53(4): 620-635.
- [8] 景运革, 李天瑞. 一种基于关系矩阵的决策表正域约简算法[J]. 计算机科学, 2013, 40(11): 261-264.

## 作者简介:

唐定勇 男, (1972-), 博士研究生. 研究方向为制造业信息服务、信息安全和数据挖掘。

景运革(通讯作者) 男, (1970-), 博士研究生. 研究方向为数据挖掘、粒计算和粗糙集. E-mail: jyg701022@163.com.