文章编号: 1001-0920(2007)03-0268-05

一种决策表增量属性约简算法

胡 峰, 代 劲, 王国胤
(重庆邮电大学 计算机科学与技术研究所, 重庆 400065)

摘 要: 为了对动态变化的决策表进行属性约简处理,在改进的分辨矩阵的基础上,提出一种增量式属性约简算法,当决策表添加新的记录后,能快速得到新决策表的所有约简和最小约简. 此外,通过对不相容决策表的正区域的决策值和边界域对原决策表进行分解,得到了一种分布式增量属性约简模型. 仿真研究表明了算法的正确性和高效性.

关键词: 粗集; 属性约简; 增量式; 分布式
中图分类号: TP18 **文献标识码**: A

Incremental algorithms for attribute reduction in decision table

HU Feng, DAI Jin, WANG Guo-yin
(Institute of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China. Correspondent: HU Feng, E-mail: hufeng@cqupt.edu.cn)

Abstract: Incremental algorithms for attribute reduction based on modified discernibility matrix are proposed, by which minimal attribute reduction cluster of new decision table can be obtained quickly when new records are added to primary decision table. A distributed model of incremental attribute reduction is also presented by decomposing values of decision attribute of positive region and boundary region in non-tolerant decision table. The simulation experiments show the validity and effectiveness of algorithms.

Key words: Rough set; Attribute reduction; Incremental; Distributed

1 引 言

粗集(RS)理论[1]是由波兰逻辑学家 Pawlak 教授于 1982 年提出的,它能用确定的方法处理不确定知识,不需要先验知识,直接从数据中获取知识,该理论已经取得了许多研究成果[2]. 在基于粗集理论的知识获取研究中,属性约简是其中最核心的组成部分之一,属性约简的结果会对最终形成的规则产生直接的影响,许多学者已对属性约简的算法进行了大量的研究[3-5]. 但这些研究几乎都是针对静态数据的,而在实际应用中,数据库往往都是动态变化的. 因此许多研究者建议[6-8],数据库知识发现算法应该是增量式的. 增量式的规则获取算法[9-11]和增量式的属性约简算法[12-16]已经开始得到研究.

目前的增量式属性约简算法大致可分为两大类:一类是获取属性约简簇集的增量式算法[12-14];另一类是获取一个属性约简的增量式算法[15,16]. 这两类算法都使用了对象之间的分辨属性来处理,属于代数观下的属性约简方法. 文献[12-14]提出了一些增量式属性约简方法,虽然能够得到信息系统的最小约简,但只能求出绝对约简(不包含决策属性),而在实际应用中,大部分数据都是具有决策属性的. 文献[15]给出了一种基于 Skowron 分辨矩阵[3]的属性约简的增量模型,但不能保证得到一个 Pawlak 约简[15],也不能处理不相容决策表. 文献[16]中给出的算法可以处理不相容决策表,得到一个完备的 Pawlak 约简,但不能得到最小约简.

本文以改进的分辨矩阵为基础提出一种增量式的属性约简算法. 该算法可以处理不相容决策表,利用该算法能够快速地得到动态变化决策表的最小约简,并通过实验证明了算法的有效性. 最后,还给出了一种分布式增量属性约简模型.

2 粗集的基本概念

为便于叙述,这里先将粗集的一些基本概念作简单介绍,本节中的定义 1～定义 6 均出自文献

收稿日期: 2005-11-04; **修回日期**: 2006-01-06.
基金项目: 国家自然科学基金项目(60373111,60573068); 新世纪优秀人才支持计划(NCET).
作者简介: 胡峰(1978—),男,湖北天门人,讲师,硕士,从事智能信息处理等研究; 王国胤(1970—),男,重庆人,教授,博士生导师,从事智能信息处理等研究.

[2].

定义 1(决策表)　一个决策表 $S = \langle U, A, V, f\rangle$,其中 U 是对象的集合,也称为论域,$A = C \cup D$ 是属性集合,子集 C 和 D 分别称为条件属性集和决策属性集,$D \neq \varnothing$,V 是属性值的集合,$f: U \times R \to V$ 是一个信息函数,它指定了 U 中每个对象 x 的属性值.

定义 2(不分明关系)　给定决策表 $S = \langle U, A, V, f\rangle$,对于每个属性子集 $B \subseteq R$,定义一个不分明关系 $\mathrm{IND}(B)$,即

$$\mathrm{IND}(B) = \{(x, y) \mid (x, y) \in U \times U, \forall b \in B(b(x) = b(y))\},$$

显然不分明关系是一种等价关系.

定义 3(条件分类和决策分类)　给定决策表 $S = \langle U, C \cup D, V, f\rangle$,$C$ 和 D 分别为决策表的条件属性集和决策属性集,$U \mid \mathrm{IND}(C)$ 和 $U \mid \mathrm{IND}(D)$ 分别为论域 U 在属性集 C 和 D 上形成的划分,条件分类定义为 $E_i \in U \mid \mathrm{IND}(C)$ $(i = 1, \cdots, m,\ m$ 为条件分类的个数);决策分类定义为 $X_j \in U \mid \mathrm{IND}(D)$ $(j = 1, \cdots, n,\ n$ 为决策分类的个数).

定义 4(相对正区域)　设 U 为一个论域,P, Q 为定义在 U 上的两个等价关系簇,Q 的 P 正区域记为 $\mathrm{POS}_P(Q)$,并定义为 $\mathrm{POS}_P(Q) = \bigcup_{X \in U/Q} P\_(X)$.

定义 5　设 U 为一个论域,P, Q 是定义在 U 上的两个等价关系簇,若 $\forall r \in P$ 都是 P 中相对于 Q 必要的,则称 P 为相对于 Q 独立的.

定义 6(相对约简)　设 U 为一个论域,P, Q 是定义在 U 上的两个等价关系簇,若 P 的 Q 独立子集 $S \subset P$ 有 $\mathrm{POS}_S(Q) = \mathrm{POS}_P(Q)$,则称 S 为 P 的 Q 约简.

在 Rough 集理论中,Pawlak 定义了两种信息系统的约简:绝对约简(不包含决策属性)和相对约简(包含决策属性).本文中所得到的约简都属于相对约简,或称为约简.

3　增量式属性约简算法

给定决策表 $S = \langle U, A, V, f\rangle$,Skowron 给出了信息系统的分辨矩阵[3],Hu 给出了修改后的决策表分辨矩阵 $M: = [C_{ij}]\,(i = 1, \cdots, \mid U \mid, j = 1, \cdots, \mid U \mid)$,$C_{ij} = \{a \in C: f(x_i, a) \neq f(x_j, a) \wedge f(x_i, D) \neq f(x_j, D), x_i, x_j \in U\}$ [5]. 然而,Hu 给出的分辨矩阵的定义在不相容决策表的处理中是不完备的[17,18]. 文献[17]给出了一个改进的分辨矩阵,本文则给出了一个与文献[17]等价的分辨矩阵,根据新定义下的分辨矩阵,可以进行增量式属性约简处理.

3.1　增量式属性约简原理

在决策表 S 中,论域 U 可以分成正区域 $\mathrm{POS}_C(D)$ 和边界域 $\mathrm{BN}_C(D)$. 一般说来,决策表的决策属性集合只包含一个元素,本文中也用 $\{d\}$ 来表示决策属性集 D. 假定 V_d 表示 $\mathrm{POS}_C(D)$ 中个体对象决策属性的属性值集合,即 $\forall x \in \mathrm{POS}_C(D)$, $\exists d_i \in V_d \wedge f(x, d) = d_i, i = 1, \cdots, \mid V_d \mid$ (为了描述方便,$f(x, d)$ 也可以表示成 $d(x)$).

将决策表 S 的论域 U 按 $\mathrm{POS}_C(D)$ 和 $\mathrm{BN}_C(D)$ 排列,并且对 $\mathrm{POS}_C(D)$ 按决策值 $f(x, d)$ 分类排列,则可以将 U 分成 $(\mid V_d \mid + 1)$ 部分,记为 $U_{d_1}, U_{d_2}, \cdots, U_{d\mid V_d\mid}, U_{\mathrm{BN}}(d_1, d_2, \cdots, d_{\mid V_d\mid} \in V_d)$. 那么可以给出与文献[17]等价的改进分辨矩阵的表示

$$M_1 = [C_{ij}],$$

$$i = 1, \cdots, \mid \mathrm{POS}_C(D) \mid, j = 1, \cdots, \mid U \mid,$$

$$C_{ij} = \begin{cases} a \in C, f(x_i, a) \neq f(x_j, a) \wedge d(x_i) \neq d(x_j), \\ 1 \leq i < j \leqslant \mid \mathrm{POS}_C(D) \mid; \\ a \in C, f(x_i, a) \neq f(x_j, a), \\ 1 \leqslant i \leqslant \mid \mathrm{POS}_C(D) \mid, \mid \mathrm{POS}_C(D) \mid < j \leqslant \mid U \mid; \\ \varnothing, \text{其他}. \end{cases}$$

命题 1(基于改进分辨矩阵的约简)　假定 M_1 是决策表 $S = \langle U, A, V, f\rangle$ 的分辨矩阵,则 $P \subset C \wedge P \neq \varnothing$ 是一个约简当且仅当 $\forall a \in M_1 \wedge a \neq \varnothing \Rightarrow a \cap P \neq \varnothing$ 且 P 是独立的.

设 $F: 2^C \to 2^C$ 是对分辨矩阵 M_1 中元素进行析取和合取运算的布尔函数,利用 F 可得到 M_1 的属性约简簇集 $R(R \in 2^C)$ 的表达式 $R = F(M_1)$.

引理 1　设 R 是决策表 S 的属性约简簇集的范式表达(析取和合取范式),M_1 是根据 S 的正区域的决策值和边界域分类后得到的分辨矩阵,F 是一个进行析取和合取运算的布尔函数,则 $R = F(M_1)$.

证明　根据 M_1 的定义容易得证. □

设 $B^v_{d_i}\,(i = 1, \cdots, \mid V_d \mid)$ 分别表示 $\mathrm{POS}_C(D)$ 中决策属性值为 d_i 的分辨矩阵元素布尔表达式,即 $B^v_{d_i} = \wedge\wedge\vee C_{ij}\,(1 \leqslant i \leqslant \mathrm{POS}_C(D), 1 \leqslant j \leqslant \mid U \mid, x_i \in U_{d_i}, x_j \in U/U_{d_i})$.

定理 1　$R = F(M_1) = \wedge B^v_{d_i}\,(i = 1, \cdots, \mid V_d \mid)$.

证明　根据 M_1 的定义容易得证. □

设 $S = \langle U, C \cup D, V, f\rangle$ 是决策表,W 是论域全集,$x(x \in W \wedge x \notin U)$ 是一个新的个体,R 是 S 的属性约简簇集,$S_1 = \langle U \cup \{x\}, C \cup D, V, f\rangle$ 是新的决策表,$POS'_C(D)$ 和 BN′ 分别表示 S_1 的正区域和边界域,则 x 与 U 在 S_1 中构成的关于 x 的分辨矩阵元素的范式表达可以表示为

$$B_1 = \begin{cases} \vee\{a \in C, f(x,a) \neq f(x_i,a), \\ 1 \leq i \leq |U|, x_i \in U/U_{d(x)}\}, x \in \mathrm{POS}_C'(D); \\ \vee\{a \in C, f(x,a) \neq f(x_i,a), \\ 1 \leq i \leq |\mathrm{POS}_C(D)|, x_i \in \mathrm{POS}_C(D)\}, x \in \mathrm{BN}. \end{cases}$$

定理2 设 $S = \langle U, C \cup D, V, f\rangle$ 是决策表，W 是论域全集，$x(x \in W \wedge x \notin U)$ 是一个新的个体，R 是 S 的属性约简簇集，R_1 是 $S_1 = \langle U \cup \{x\}, C \cup D, V, f\rangle$ 的属性约简簇集，布尔表达式 B_1 是 x 与 U 在 S_1 中构成的关于 x 的分辨矩阵元素的范式表达，则有 $R_1 = R \wedge B_1$.

证明 根据定理1和 B_1 的定义容易得证. □

3.2 增量式属性约简算法

根据决策表 S 的决策属性值和正区域与边界域的定义，得到了 S 中论域的一个划分 $S = U_{d_1} \cup U_{d_2} \cup \cdots \cup U_{d|V_d|} \cup U_{\mathrm{BN}}$. 在此基础上，利用定理2可以给出 S 的增量式属性约简算法.

算法1 增量式属性约简算法

输入：决策表 $S = \langle U, C \cup D, V, f\rangle$，$S$ 的属性约简簇集 R，新加入的对象集合 Add.

输出：新决策表 $S_1 = \langle U \cup \mathrm{Add}, C \cup D, V, f\rangle$ 的属性约简簇集 R_1.

Step1：将决策表 S 的论域 U 按 $\mathrm{POS}_C(D)$ 和 $\mathrm{BN}_C(D)$ 排列，并且对 $\mathrm{POS}_C(D)$ 按决策值 $f(x,d)$ 分类排列，则可以将 U 分成 $(|V_d|+1)$ 部分，分别记为 $U_{d_1}, U_{d_2}, \cdots, U_{d|V_d|}, U_{\mathrm{BN}}$.

Step2：$R_1 = R, B = \varnothing$.

Step3：For $i = 1$ to $|\mathrm{Add}|$ Do

从 Add 中取出一条记录 Add_i，计算 Add_i 与 U 在 $S' = \langle U \cup \{\mathrm{Add}_i\}, C \cup D, V, f\rangle$ 中构成的关于 Add_i 的分辨矩阵元素的范式表达 B_i. $R_1 = R_1 \wedge B_i$，$U = U \cup \{\mathrm{Add}_i\}$，按下列过程更新 $U_{d_1}, U_{d_2}, \cdots, U_{d|V_d|}, U_{\mathrm{BN}}$：

1) 若 $d(\mathrm{Add}_i) \notin V_d$，则进行如下处理：$V_d = V_d \cup d(\mathrm{Add}_i)$，$U_{d(\mathrm{Add}_i)} = U_{d(\mathrm{Add}_i)} \cup \{\mathrm{Add}_i\}$，转 Step3.

2) 若 $\exists x \in U$ 满足 $f(x,C) = f(\mathrm{Add}_i, C) \wedge d(x) \neq d(\mathrm{Add}_i)$，则进行如下处理：$U_{\mathrm{BN}} = U_{\mathrm{BN}} \cup \{\mathrm{Add}_i\}$，若 $x \in U_{\mathrm{BN}}$，转 Step3；否则，$U_{d(x)} = U_{d(x)}/\{x\}$，转 Step3.

3) $U_{d(\mathrm{Add}_i)} = U_{d(\mathrm{Add}_i)} \cup \{\mathrm{Add}_i\}$，转 Step3.

Loop

Step4：return R_1，Stop.

3.3 算法的时空复杂度分析

令原始决策表中记录数为 n，属性个数为 m，新添加的记录数为 p，属性约简簇集的势为 CM，极小相对约简簇集的势为 PM，则算法1的时间复杂度为 $t_1 = O(\mathrm{CM}^2 \times p \times (n+p) \times m^2)$，空间复杂度为 $s_1 = O(m \times (n+p))$.

4 实验测试

为验证本文方法的有效性，这里使用了 RIDAS 系统[19] 作为工具. 此外，为了考察文中算法与已有的增量属性约简算法的优劣性，选择了文献[16]中的算法进行测试比较，记作算法 a（在文献[12-16]中，只有[16]的算法能处理不相容决策表）. 实验的硬件测试环境是：CPU：P4 2.6 GHz，内存：256 MB，操作系统：WindowsXP.

具体测试过程如下：1) 利用基于分辨矩阵的属性约简算法得到原决策表的相对属性约简簇集；2) 对添加的记录集使用算法1进行增量式计算，并与非增量式方法相比较；3) 从测试过程第一步得到的相对属性约简簇集中选择一个最小约简作为算法 a 的初始约简，对添加的记录集使用算法 a 进行增量式计算.

4.1 UCI 数据库测试

选用 UCI 数据库中的数据集 Heart_c_ls，Pima_India，Crx_bq_ls，Liver_disorder 和 Abalone 为测试对象. 从这5个数据集中随机抽取 80 % 的记录构成原决策表，剩下的 20 % 作为新添加的记录集. 各种算法的比较结果见表1. 其中：属性数表示测试记录集中的条件属性的数目；T 表示算法的运行时间(s)，运行时间为0的表示小于 0.001 s；N 表示所有约简的簇集的势；MN 表示所得的属性约简结果中最小约简包含的属性数目；CMN 表示约简结果中最小约简簇集的势；N_a 表示算法 a 得到的约简

表1 增量式属性约简算法测试结果（UCI 数据库）

数据集	属性数	记录数	基于分辨矩阵约简				算法1				算法 a	
			T	MN	N	CMN	T	MN	N	CMN	T	N_a
Heart_c_ls	13	303	0.26	9	1	1	0.01	9	1	1	0.01	9
Crx_bq_ls	15	690	2.704	6	65	1	7.48	6	65	1	0.05	6
Pima_India	8	738	0.701	5	1	1	0	5	1	1	0.04	5
Liver_disorder	6	1 260	0.14	5	1	1	0	5	1	1	0.03	5
Abalone	8	4 177	63.772	6	1	1	0	6	1	1	1.542	6

的条件属性个数(这些符号在表 2 和表 3 中也有相同的含义).

4.2　自定义数据集测试

为充分测试算法的性能,自定义一组数据进行测试. 随机生成 5 个数据集:DS_1,DS_2,DS_3,DS_4 和 DS_5,它们具有 12 个条件属性和一个决策属性,分别包含记录 1 000 条,2 000 条,3 000 条,4 000 条和 5 000 条. 其中前 90 % 的记录作为原始决策表,后 10 % 的记录作为添加的数据集. 按如下方式控制它们的生成:它们的前 7 个条件属性在 0 ～ 9 上随机取值,后 5 个条件属性在 0 ～ 1 上随机取值;决策属性在 0 ～ 4 上随机取值. 测试比较结果见表 2.

从表 1 和表 2 可以看出,算法 1 在时间性能上要优于非增量式方法. 由于算法 a 只计算一个属性约简,一般说来,算法 a 的运算时间会小于算法 1. 但分析表 1 和表 2,可以发现一个有趣的结果:当决策表的属性约简簇集的势较小时,算法 1 的性能要好于算法 a;当决策表约简簇集的势较大时,算法 1 的性能较差. 出现这种结果的主要原因是:算法 1 的时间复杂度与决策表属性约简簇集的势呈平方关系;而在算法 a 中,进行了多次论域中对象的比较运算,此外,算法 a 也进行了消除冗余属性的运算.

表 2　增量式属性约简算法测试结果(自定义数据集)

数据集	属性数	记录数	基于分辨矩阵约简				算法 1				算法 a	
			T	MN	N	CMN	T	MN	N	CMN	T	N_a
DS_1	12	1 000	7.27	5	79	1	6.82	5	79	1	0.21	6
DS_2	12	2 000	26.047	6	41	1	4.006	6	41	1	0.35	6
DS_3	12	3 000	60.216	7	31	8	3.054	7	31	8	1.022	7
DS_4	12	4 000	102.658	7	30	1	3.025	7	30	1	1.412	7
DS_5	12	5 000	194.86	7	12	4	1.081	7	12	4	3.195	8

5　分布式增量属性约简

随着 Internet 技术的迅猛发展,分布式处理技术越来越受到重视. 在属性约简过程中,当数据量很大时,如果能够充分地利用分布式处理技术,将会大大提高其处理速度. 通过对分辨矩阵和算法 1 的分析,本文提出了一种分布式增量属性约简模型.

5.1　分布式增量式属性约简算法

利用决策表 S 的正区域的决策属性值和边界域,可以将 S 分解成多个子决策表,分别分布到不同的 Web 端,当新的记录加入 S 时,可以在所有的 Web 端同时进行约简处理. 处理完之后,再进行约简簇集结果合并,就可以得到新决策表的约简簇集. 根据定理 1 和定理 2 可以给出以下算法:

算法 2　分布式增量式属性约简算法

输入:决策表 $S=\langle U, C\cup D, V, f\rangle$,$S$ 的约简簇集 R,新加入的对象集合 Add.

输出:新信息系统 $S_1=\langle U\cup \mathrm{Add}, C\cup D, V, f\rangle$ 的属性约简簇集 DR.

Step1: 将决策表 S 的论域 U 按 $\mathrm{POS}_C(D)$ 和 $\mathrm{BN}_C(D)$ 排列,并且对 $\mathrm{POS}_C(D)$ 按决策值 $f(x,d)$ 类排列,则可将 U 分成($|V_d|+1$)部分,分别记为 $U_{d_1}, U_{d_2}, \cdots, U_{d|V_d|}, U_{\mathrm{BN}}$. 并将记录 $U_{d_1}, U_{d_2}, \cdots, U_{d|V_d|}, U_{\mathrm{BN}}$ 分别分布到不同的 Web 端:$W_{d_1}, W_{d_2}, \cdots, W_{d|V_d|}, W_{\mathrm{BN}}$.

Step2: DR = ∅, $B_{d_i}^V=\varnothing (1\le i\le |V_d|)$.

Step3:加入新对象集合,并进行分布增量约简.

For i = 1 to | Add | Do

从对象集合 Add 中取出一条记录 Add_i,$U = U\cup\{\mathrm{Add}_i\}$.

1) For j = 1 to | U | Do

If $\exists x_j\in \mathrm{BN}_C(D), f(x_j,C)=f(\mathrm{Add}_i,C)$ 满足 $f(x_j,C)=f(\mathrm{Add}_i,C)$, Then $U_{\mathrm{BN}}=U_{\mathrm{BN}}\cup\{\mathrm{Add}_i\}$,转 Step3;

Loop

2) If $f(\mathrm{Add}_i,d)\notin V_d$, Then

$V_d = V_d\cup\{f(\mathrm{Add}_i,d)\}$;

$\mathrm{POS}_C(D) = \mathrm{POS}_C(D\cup\{\mathrm{Add}_i\})$;

新增加一 Web 端 $W_{d|V_d|}$,将 Add_i 分布到 $W_{d|V_d|}$;计算各 Web 端与 Add_i 的分辨矩阵的范式表达 $B_k(1\le k\le |V_d|-1)$;$B_{d_i}^V = B_{d_i}^V\cup B_k$;在 $W_{d|V_d|}$ 端计算 Add_i 与 U 的分辨矩阵的范式表达 $B_{d|V_d|}$;$B_{d|V_d|}^V = B_k$;转 Step3;

End If

3) For j = 1 to | U | Do

If $\exists x_j\in U, f(x_j,C)=f(\mathrm{Add}_i,C)\wedge d(x_j)\neq d(\mathrm{Add}_i)$, Then

$U_{d(x_j)} = U_{d(x_j)} - \{x_j\}$,

$U_{\mathrm{BN}} = U_{\mathrm{BN}}\cup\{x_j\}\cup\{\mathrm{Add}_i\}$;

End If

Loop

4) 在各 Web 端计算 Add_i 与 $\mathrm{POS}_C(D)$ 中对象的分辨矩阵的范式表 $B_k(1\le k\le |V_d|)$,并更新各

Web 端的范式表达 $B_{d_i}^V = B_{d_i}^V \cup B_k$,转 Step3;

Loop

Step4: 在各 Web 端计算 $B_{d_i}^V$.

Step5: 根据定理 1,可得 $DR = R \wedge B_{d_1}^V \wedge B_{d_2}^V \wedge \cdots \wedge B_{d|V_d|}^V$.

Step6: return DR, Stop.

5.2 算法时空复杂度分析与测试比较

令原始决策表中记录数为 n,条件属性个数为 m,新添加的记录数为 p,属性约简簇集的势为 CM,决策属性值集合的势为 k,则算法 2 的时间复杂度为 $t_2 = O(CM^2 \times p \times (n+p) \times m^2/k)$,空间复杂度为 $s_2 = O(m \times (n+p))$.

为了分析算法 2 与算法 1 以及算法 a 的性能差异,进行了一组测试,在此测试过程中,没有考虑不同 Web 端之间的通信代价. 测试环境是:CPU:P4 2.6 GHz,内存:256 MB,操作系统:WindowsXP. 测试过程和测试数据与 4.2 节相同,测试比较结果见表 3.

在实验中发现,在算法 2 中,最耗时的计算在 Step4,基本要耗费总时间的 60 % 左右,而 Step5 耗费的时间非常小. 算法 2 由于采用了分布式处理,可以在 Step4 中进行分布并行处理,当决策表决策属性分类较多时,利用算法 2 可以提高增量式属性约简的执行效率.

表 3 分布式增量属性约简测试结果(自定义数据集)

数据集	属性数	记录数	算法 2				算法 1				算法 a	
			T	MN	N	CMN	T	MN	N	CMN	T	N_a
DS_1	12	1 000	4.306	5	79	1	6.82	5	79	1	0.21	6
DS_2	12	2 000	3.282	6	41	1	4.006	6	41	1	0.35	6
DS_3	12	3 000	1.813	7	31	8	3.054	7	31	8	1.022	7
DS_4	12	4 000	1.801	7	30	1	3.025	7	30	1	1.412	7
DS_5	12	5 000	0.942	7	12	4	1.081	7	12	4	3.195	8

6 结 语

增量式学习是人工智能领域一个重要的问题,属性约简是粗集理论研究中最核心的工作之一. 本文在改进的分辨矩阵的基础上,着重讨论了增量式属性约简算法,并进行了实验测试. 在已知原决策表属性约简簇集的基础上,算法 1 能够增量地求出新决策表的相对属性约简簇集,实验测试也证明了算法 1 的有效性. 此外,本文还给出了一个分布式增量属性约简算法,根据决策表正区域的决策属性值对原系统进行划分,得到了一个较好的分布式增量属性约简模型.

参考文献(References)

[1] Pawlak Z. Rough set [J]. Int J of Computer and Information Sciences, 1982, 11: 341-356.

[2] 王国胤. Rough 集理论与知识获取[M]. 西安: 西安交通大学出版社, 2001.
(Wang G Y. Rough set theory and knowledge acquisition [M]. Xi'an: Xi'an Jiaotong University Press, 2001.)

[3] Skowron A, Rauszer C. The discernibility functions matrics and functions in information systems [C]. Intelligent Decision Support — Handbook of Applications and Advances of the Rough Sets Theory. Dordrecht: Kluwer Academic Publisher, 1992: 331-362.

[4] 王国胤, 于洪, 杨大春. 基于条件信息熵的决策表约简[J]. 计算机学报, 2002, 25(7): 759-766.
(Wang G Y, Yu H, Yang D C. Decision table reduction based on conditional information entropy[J]. Chinese J of Computer, 2002, 25(7): 759-766.)

[5] Hu X H, Cercone N. Learning in relational database: A rough set approach [J]. Int J of Computational Intelligence, 1995, 11(2): 323-338.

[6] Fayyad U M, Piatetsky-Shapiro L, Smyth P, et al. Advances in knowledge discovery and data mining[C]. AAAI Press. Mcnlo Park: MIT Press, 1996.

[7] Cercone V, Tsuchiya M. Luesy editors introdunction [J]. IEEE Trans on Knowledge and Data Engineering, 1993, 5(6): 901-902.

[8] Piatetsky-Shapiro L, Frawley W J. Knowledge discovery in database[C]. AAAI Press. Menlo Park: MIT Press, 1991.

[9] Zheng Z, Wang G Y, Wu Y. A rough set and rule tree based incremental knowledge acquisition algorithm[C]. Lecture Notes In Artificial Intelligence 2639. Chongqing: Springer-Verlag, 2003: 122-129.

[10] 於东军, 王士同, 杨静宇. 一种增量式规则提取算法[J]. 小型微型计算机系统, 2004, 25(1): 79-81.
(Yu D J, Wang S T, Yang J Y. An incremental rule extractionalgorithm[J]. Mini-micro Systems, 2004, 25(1): 79-81.)

[11] Ziarko W, Shan N. Data-based acquisition and incremental modification classfication rules [J]. Computational Intelligence, 1995, 11(2): 357-370.

(下转第 277 页)

现鲁棒的自定位,而从表 2 可以看出,常规粒子滤波方法在粒子数目为 2 500 左右就已经发散了. 因此,改进后算法的收敛性相比于常规方法有一定的提高.

5 结　　语

在利用常规粒子滤波方法进行系统状态预估时,通常粒子集数目不能太大,否则系统的实时性很差. 另一方面,如果粒子集的数目太小,则系统的鲁棒性将会降低,容易受到粒子贫乏现象的影响. 特别是在观测量较准确或似然概率位于先验概率尾部的情况下,常规粒子滤波器的预估性能很差. 本文通过分析常规粒子滤波方法存在问题的原因,将粒子群优化的思想引入粒子滤波中. 通过将最新的观测值引入采样分布中,并利用粒子群优化算法对采样过程进行优化,使得采样分布向后验概率较高的区域运动,从而避免了粒子贫乏现象的产生,同时提高了状态预估的精度. 此外,PSOPF 还可以解决系统初始状态未知情况下的预估问题,并可明显地降低所需粒子数,提高系统的鲁棒性. 实验结果表明了 PSOPF 算法的有效性.

参考文献(References)

[1] Bogdan K. Finding location using a particle filter and histogram matching [C]. Proc of Artificial Intelligence and Soft Computing. Poland: Springer, 2004:786-791.

[2] Doucet A. On sequential simulation based methods for Bayesian filtering [J]. Statistics and Computing, 1998, 10(3): 197-208.

[3] Thrun S. Particle filters in robotics [C]. Proc of Uncertainty in AI. San Francisco: Morgan Kaufmann Publishers, 2002: 511-518.

[4] Carpenter J, Clifford P, Fernhead P. An improved particle filter for non-linear problems [R]. Oxford: University of Oxford, 1997.

[5] Van M R, Doucet A. The unscented particle filter[R]. Cambridge: Cambridge University, 2000.

[6] Clapp T C. Statistical methods for the processing of communication data [D]. Cambridge: University of Cambridge, 2000.

[7] Ronghua L, Bingrong H. Coevolution based adaptive monte Carlo localization [J]. Int J of Advanced Robotic Systems, 2004, 1(3): 183-190.

[8] Peter T, Csaba S. LS-N-IPS: An improvement of particle filters by means of local search [C]. Proc Nonlinear Control Systems. Petersburg, 2001: 715-719.

[9] Jun S L, Rong C, Tanya L. A theoretical framework for sequential importance sampling and resampling [C]. Sequential Monte Carlo in Practice. Doucet: Springer-Verlag, 2001: 225-246.

[10] Kennedy J, Eberhart R. Particle swarm optimization [C]. Proc of the IEEE Int Conf on Neural Networks, Piscataway: IEEE Service Center, 1995: 1941-1948.

[11] Krohling R A. Gaussian swarm: A novel particle swarm optimization algorithm[C]. Proc of the IEEE Conf on Cybernetics and Intelligent Systems. Singapore, 2004: 372-376.

[12] Ioannis R. A particle filter tutorial for mobile robot localization [R]. Montredal, Quebec: McGill University, 2004.

(上接第 272 页)

[12] Susmaga R. Experiments in incremental computation of reducts[C]. Rough Sets in Data Mining and Knowledge Discovery. Berlin: Springer-Verlag, 1998.

[13] 刘宗田. 属性最小约简的增量式算法[J]. 电子学报, 1999, 27(11): 96-98.
(Liu Z T. An incremental arithmetic for the smallest reduction of attributes [J]. Acta Electronica Sinica, 1999, 27(11): 96-98.)

[14] Bazan B, Nguyen Hung Son, Nguyen Sinh Hoa. Rough set algorithms in classification problem [C]. Rough Set Methods and Applications. Heidelberg: Physica- Verlag, 2000: 49-88.

[15] Wang Jue, Wang Ju. Reduction algorithms based on discerniblility matrix: The ordered attributed method [J]. J of Computer Science and Technlology, 2001, 11(6): 489-504.

[16] Hu F, Wang G Y, Huang H, et al. Incremental attribute reduction based on elementary sets [C]. Lecture Notes In Artificial Intelligence 3641. Regina: Heidelberg, Physica-Verlag, 2005: 185-193.

[17] 叶东毅, 陈昭炯. 一个新的差别矩阵及其求核方法[J]. 电子学报, 2002, 30(7): 1086-1088.
(Ye D Y, Chen Z Y. A new discernibility matrix and the computation of a core[J]. Acta Electronica Sinica, 2002, 30(7): 1086-1088.)

[18] 王国胤. 决策表核属性的计算方法[J]. 计算机学报, 2003, 26(5): 611-615.
(Wang G Y. The computation method of core attribute in decision table[J]. Chinese J of Computer, 2003, 26(5): 611-615.)

[19] Wang G Y, Zheng Z, Zhang Y. RIDAS —A rough set based intelligent data analysis system[C]. The 1st Int Conf on Machine Learning and Cybernetics. Beijing, 2002: 646-649.