文章编号: 1001-0920(2006)09-0991-05

一种基于垂直分布的多决策表全局属性核求解算法

杨 明, 杨 萍
(南京师范大学 数学与计算机科学学院, 南京 210097)

摘 要: 针对分布式环境下粗糙集理论研究的重要性, 引入全局决策表和局部决策表的概念, 并提出一种基于垂直分布的多决策表全局属性核求解算法. 该算法可充分利用求得的局部不一致对象和局部差别矩阵, 有效提高垂直分布的多决策表全局属性核求解效率. 分析结果表明本文算法是有效的, 可行的.

关键词: 粗糙集; 差别矩阵; 全局属性核; 多决策表
中图分类号: TP311 **文献标识码**: A

An Algorithm over Vertically Partitioned Multi-decision Table for Computing Global Attribute Core

YANG Ming, YANG Ping
(School of Mathematics and Computer Science, Nanjing Normal University, Nanjing 210097, China. Correspondent: YANG Ming, E-mail: m.yang@njnu.edu.cn)

Abstract: The concepts of global decision table and local decision table are introduced. And an algorithm over vertically partitioned multi-decision table for computing global attribute core is presented. The algorithm can efficiently utilize every local inconsistent objects and every local discernibility matrix at each site, so the efficiency for computation of global attribute core can be improved efficiently. Analysis results show the proposed algorithm is effective and efficient.

Key words: Rough set; Discernibility matrix; Global attribute core; Multi-decision table

1 引 言

随着数据库技术和网络技术的快速发展, 各类企事业单位十分重视信息化建设, 逐步构成分布式环境下的多数据库系统, 并存储大量的历史数据和运行数据. 随着计算机技术深入应用的需要, 企事业单位的决策层迫切需要将这些分布数据转换成有用的信息和知识, 为商务管理、生产控制、市场分析、工程设计、科学探索和网络安全检测等提供决策依据, 因此分布式环境下的分类、关联等算法的研究受到研究者的广泛关注, 并取得了一些进展[1~4].

粗糙集(RS)作为一种新的处理不精确、不完全和不相容知识的数学理论[5], 是分类规则获取的重要方法. 近年来该理论在机器学习、数据挖掘和网络入侵检测等多个领域得到了广泛的应用[6~8], 但在现实应用中, 很多决策应用问题需要通过分布式环境下的各参与方的协同工作来完成, 因此, 进行分布式环境下的粗糙集理论研究具有重要的理论和现实意义.

在粗糙集理论研究中, 属性约简是分类规则获取的关键步骤[9], 而较多经典的属性约简算法都是从属性核开始的, 求属性核成为属性约简求解的关键步骤. 但现有高效的属性核求解算法主要针对单个决策表的属性核求解[10~13], 针对分布式环境下的多个决策表全局属性核求解算法研究还少有报道.

收稿日期: 2005-07-06; **修回日期**: 2005-10-24.
基金项目: 国家自然科学基金项目(70371015); 江苏省自然科学基金项目(BK2005135); 江苏省高校自然科学研究基金项目(05KJB520066).
作者简介: 杨明(1964—), 男, 安徽宁国人, 教授, 博士, 从事数据挖掘、知识发现等研究; 杨萍(1967—), 女, 安徽宁国人, 副教授, 从事管理决策、粗糙集理论与应用等研究.

对于垂直分布的多决策表全局属性核求解问题, 一种简单方法就是在某个中心站点将各局部决策表通过连接操作得到单个决策表, 然后直接应用现有的属性核求解算法, 但这种处理方法效率较低. 为此, 本文引入全局决策表和局部决策表的概念, 并提出一种基于垂直分布的多决策表全局属性核求解算法, 该算法可充分利用求得的局部不一致对象和局部差别矩阵, 有效提高垂直分布的全局属性核求解的效率. 算法分析结果表明, 本文的算法是有效的和可行的.

2 相关概念及结论

在本文的模型中, 多个决策表是垂直划分的. 设有 t 个站点, $S_1, S_2, \cdots, S_t$, 相应的成员决策表 DT_i (或局部决策表) 的属性集分别为 $C_1 \cup D, C_2 \cup D, \cdots, C_t \cup D$, $\bigcap_{i=1}^{t} C_i = \varnothing$ ($\varnothing$ 表示空集), 各局部决策表具有相同的对象集 U 且均隐含一个对象标识属性, 通过该属性可将各局部决策表连接成一个单决策表.

为便于叙述, 设条件属性集合 C 的值域为有限离散集合, 用 $|\cdot|$ 表示集合的基. 不失一般性, 将各局部决策表连接后得到的决策表记为 $\mathrm{DT} = \langle U, C \cup D, V, f \rangle$, $C = \bigcup_{i=1}^{t} C_i$, $m = \max_{1 \le i \le t}(|C_i|)$, 并假设惟一的决策属性 D, $D = 1, 2, \cdots, k$. 由 D 导出的等价类构成 U 的一个划分: $\{\Psi_1, \Psi_2, \cdots, \Psi_k\}$, 其中: $\Psi_i = \{x \in U: f(x, D) = i\}$, $i = 1, \cdots, k$, U 中的对象个数为 n.

2.1 粗糙集的相关概念

粗糙集理论的要点是将分类与知识联系在一起, 并用等价类关系形式化表示分类. 可理解为: 知识是使用等价类 R 对离散空间 U 的划分, 记为 $U/R = \{X_1, X_2, \cdots, X_n\}$, X_i 为 U/R 的等价类.

定义 1 全局决策表 DT 是一个 4 元组 $\langle U, C \cup D, V, f \rangle$, 其中: U 是一组对象的非空有限集合, 称为论域, 设有 n 个对象, 则 U 可表示为 $U = \{x_1, x_2, \cdots, x_n\}$, C 为条件属性集, D 为决策属性集, $V = \bigcup_{a \in (C \cup D)} V_a$, V_a 为属性 a 的值域集, f 是 $U \times (C \cup D) \to V$ 的映射.

定义 2 在站点 S_i ($i = 1, 2, \cdots, t$), 局部决策表 DT_i 是一个 4 元组 $\langle U, C_i \cup D, V, f \rangle$. 其中: C_i 为条件属性集, D 为决策属性集, $V = \bigcup_{a \in (C_i \cup D)} V_a$, V_a 为属性 a 的值域集, f 是 $U \times (C_i \cup D) \to V$ 的映射.

在全局决策表或局部决策表中, 若一些对象具有相同的条件属性而具有不同的分类, 则称这类对象是全局不一致的或局部不一致的, 否则为全局一致的或局部一致的; 若 U 中的两个不同的对象 x 和 y 在条件属性集 C 上具有相同的条件属性值而具有不同的分类, 则称 x 和 y 为全局不一致的, 否则称 x 和 y 为全局一致的; 若在站点 S_i, U 中的两个不同的对象 x 和 y 在条件属性集 C_i 上具有相同的条件属性而具有不同的分类, 则称 x 和 y 为局部不一致的, 否则称 x 和 y 为局部一致的.

本文称所有对象均全局一致的全局决策表为全局一致决策表, 否则称为全局不一致决策表; 称所有对象均局部一致的局部决策表为局部一致决策表, 否则称为局部不一致决策表.

定义 3 设 $X \subseteq U$ 为论域的一个子集, $P \subseteq C$, X 的关于 P 的全局下近似为 $\underline{P}X(C) = \{x \in U: [x]_P \subseteq X\}$, 其中 $[x]_P = \{y \in U \mid \forall a \in P, f(x, a) = f(y, a)\}$.

定义 4 设 $P \subseteq C$, 对划分 $\{\Psi_1, \Psi_2, \cdots, \Psi_k\}$ 的 P-全局近似精度为

$$\gamma_{(P,S)} = \sum_{i=1}^{k} |\underline{P}\Psi_i(U)| / |U|.$$

定义 5 设 $P \subseteq C$, 若 $\gamma_{(P,S)} = \gamma_{(C,S)}$, 且不存在 $R \subset P$, 使得 $\gamma_{(R,S)} = \gamma_{(C,S)}$, 则称 P 为 C 的一个 (相对于决策属性 D 的) 全局属性约简. 所有 C 的全局属性约简的交称为 C 的全局属性核, 记为 $\mathrm{Core}(C)$.

性质 1 DT 是全局不一致的当且仅当在任意局部站点 DT_i ($i = 1, \cdots, t$) 是局部不一致的.

性质 2 若在站点 S_k ($1 \le k \le t$), U 中的对象 x 和 y 是局部一致的, 则对象 x 和 y 是全局一致的.

性质 3 DT 是全局一致的当且仅当对 U 中的任意对象 x 和 y, 存在站点 S_i ($1 \le i \le t$) 使得 x 和 y 在其上是一致的.

2.2 基于差别矩阵的单决策表属性核求解方法

对于单个决策表, 即本文中的局部决策表数 t 为 1, 求得全局属性核 (这里, 全局属性核和局部属性核一致) 如下[12]:

定义 6 对给定的单个决策表 DT, 定义差别矩阵 $M = \{m_{ij}\}$ 为

$$m_{ij} = \begin{cases} \{a \in C: f(x_i, a) \ne f(x_j, a)\}, \\ f(x_i, D) \ne f(x_j, D), x_i \in U_1, x_j \in U_1; \\ \{a \in C: f(x_i, a) \ne f(x_j, a)\}, \\ x_i \in U_1, x_j \in U_2^1; \\ \varnothing, \text{其他} \end{cases} \tag{1}$$

其中

$$U_1 = \bigcup_{i=1}^{h} \underline{C}\Psi_i(U), U_2 = U - U_1,$$

$$U_2^1 = \{y \in U_2 \mid \text{不存在 } x \in U_2$$

使得 $f(x,a) = f(y,a)$

且 $f(x,D) \neq f(y,D), \forall a \in C\}$.

定理 1　对于单个决策表 DT, 若记 $DM(C, M) = \{m_{ij} | m_{ij} \in M$ 且 m_{ij} 为单个属性$\}$, 则有 $DM(C,M) = Core(C)$, 即当且仅当某个 m_{ij} 为单个属性时, 该属性属于核 $Core(C)$.

定义 6 和定理 1 为求解单个决策表的属性核提供了有效的框架, 但仅适用一个局部站点的情况, 对于多个局部站点, 即对多个决策表全局属性核求解是本文的主要内容.

3　多决策表全局属性核求解算法

为便于对分布式环境下的多决策表全局属性核求解进行讨论. 首先, 讨论求解全局不一致对象集算法, 然后讨论多决策表全局属性核求解算法.

3.1　求解全局不一致对象集算法

为求解全局不一致对象, 一种简单的方法是将各局部决策表通过对象标识连接为单决策表, 然后对该单决策表进行不一致对象求解, 但这种方法的网络通讯代价高, 影响全局属性核的求解效率. 因此, 本文通过各站点上的局部不一致对象来求全局不一致对象, 无须传送各站点上的对象, 这样可有效降低网络通讯代价.

为便于叙述, 令 $\langle x, y\rangle$ 为某个站点 S_i 上的不一致对象对, 令 $INCON(i) = \{\langle x, y\rangle | \langle x, y\rangle$ 为某个站点 S_i 上的不一致对象对$\} - \{\langle x, z\rangle | \langle x, y\rangle \in INCON(i), \langle y, z\rangle \in INCON(i)$, 且 $f(x,D) \neq f(z,D)\}$; 令 $INCON = \bigcap_{i=1}^{t} INCON(i), U_2 = \{x | \langle x, y\rangle \in INCON\} \cup \{y | \langle x, y\rangle \in INCON\}, U_2^1 (= reductionobject(U2))$ 为约简后的不一致对象集合, 可按下列算法得到:

function reductionobject(U_2)

$\{U_2^1 = \varnothing$;

for each $x \in U_2$ do

if 不存在 $z \in U_2^1$ 使得 $f(x,a) = f(z,a), \forall a \in C$, then $U_2^1 = U_2^1 \cup \{x\}$; }

性质 4　对给定的站点 S_i, 若 $\langle x, y\rangle$ 和 $\langle y, z\rangle$ 均为局部不一致对象对, 且 $f(x,D) \neq f(z,D)$, 则 $\langle x, z\rangle$ 为局部不一致对象对.

定理 2　对给定的 t 局部决策表 $DT_i (i = 1, 2, \cdots, n), U_2$ 是全局决策表 DT 的所有全局不一致对象组成的集合.

证明　由性质 1 可知, 对任意两个全局不一致对象 x 和 y, 在任意站点 S_i 上有 $\langle x, y\rangle \in INCON(i)$ 成立, 其中 $i = 1, 2, \cdots, n$. 故 $\langle x, y\rangle \in \bigcap_{i=1}^{t} INCON(i)$. 反之, 若 $\langle x, y\rangle \in \bigcap_{i=1}^{t} INCON(i)$, 则 x 和 y 为全局不一致对象. 故 U_2 是全局决策表 DT 的所有全局不一致对象组成的集合. □

由定理 2, 求全局不一致对象算法如下:

算法 1　CGIC (Computing Globally Inconsistent Objects) algorithm.

输入: 有 t 个分布式站点 $S_1, \cdots, S_t$, 各站点上的局部决策表分别为 $DT_i = \langle U, C_i \cup D, V, f\rangle, i = 1, 2, \cdots, t$. 全局决策表为 $DT = \langle U, C \cup D, V, f\rangle$, 其中 $C = \bigcup_{i=1}^{t} C_i$.

输出: 全局不一致对象集 U_2 和 U_2^1.

Step 1: 并行求各局部不一致对象集合 $INCON(i), i = 1, 2, \cdots, t$

Step 2: $k = 1$; // 找局部不一致对象个数最少 // 的站点 k

for $i = 2$ to t do

if $|INCON(i)| \leq |INCON(k)|$ then $k = i$;

Step 3: 按照 $INCON(i)$ 中的对象对 $\langle x, y\rangle$ 对 $INCON(i)$ 进行升序排序;

Step 4: $U_2 = U_2^1 = \varnothing$;

for each $\langle x, y\rangle \in INCON(k)$ do

if 任意站点 $S_i (1 \leq i \neq k \leq t)$ 上均有 $\langle x, y\rangle \in INCON(i)$ then

{

if $x \notin U_2$ then $U_2 = U_2 \cup \{x\}$;

if $y \notin U_2$ then $U_2 = U_2 \cup \{y\}$;

if 不存在 $z \in U_2^1$ 使得 $f(x,a) = f(z,a), \forall a \in C$, then

$U_2^1 = U_2^1 \cup \{x\}$;

}

Step 5: 返回 U_2 和 U_2^1.

算法 1 中 Step 1 的时间复杂度为 $O(mn\log^n)$; Step 2 的时间复杂度为 $O(n)$; Step 3 的时间复杂度为 $O(mn\log^n)$; Step 4 的时间复杂度为 $O(n)$. 可见, 算法 1 的时间复杂度为 $O(mn\log^n)$.

从算法 1 的 Step 4 可以看出, 在最坏情况下, 网络通讯量为 $O(2(t-1)\min_{1\leq i\leq t}(|INCON(i)|))$, 其中 $\min_{1\leq i\leq t}(|INCON(i)|) \leq n$; 而将 t 个局部决策表合并需要的通讯量为至少为 $O(\min_{1\leq i\leq t}(|C_i|)(t-1)n)$, 且在实际应用中, 通常情况下有 $\min_{1\leq i\leq t}(|C_i|) \geq 2$.

3.2　多决策表全局属性核求解算法

算法 1 将 U 划分为全局不一致对象集合 U_2 和

全局一致对象集合$U_1 = U - U_2$，且给出全局不一致对象集合U_2的约简U_2^1. 对各局部决策表$DT_k(k = 1, 2, \cdots, t)$，类似于定义 6 得到如下定义:

定义 7 对给定的局部决策表$DT_k(k = 1, 2, \cdots, t)$，定义相应的差别矩阵$M[k] = \{m_{ij}^k\}$为

$$m_{ij}^k = \begin{cases} \{a \in C_k: f(x_i, a) \neq f(x_j, a)\}, \\ f(x_i, D) \neq f(x_j, D), x_i \in U_1, x_j \in U_1; \\ \{a \in C_k: f(x_i, a) \neq f(x_j, a)\}, \\ x_i \in U_1, x_j \in U_2^1; \\ \varnothing, \text{其他} \end{cases} \tag{2}$$

定理 3 对各局部决策表$DT_k(k = 1, 2, \cdots, t)$，若将定义 6 中的决策表DT 看成 t 个局部决策表连接得到，则$m_{ij} = \bigcup_{k=1}^{t} m_{ij}^k$.

证明 由定义 6 和定义 7 可证 □

推论 1 对各局部决策表$DT_k(k = 1, 2, \cdots, t)$，若将定义 6 中的决策表DT 看成 t 个局部决策表连接得到，则M 中的元素m_{ij}为单个属性$\{a\}$当且仅当存在 k 使得$M[k]$的元素$m_{ij}^k = \{a\}(1 \leq k \leq t)$，且$\forall s(1 \leq s \neq k \leq t)$ 有$M[s]$的元素$m_{ij}^s = \varnothing$.

定理 4 对各局部决策表$DT_k(k = 1, 2, \cdots, t)$，属性$\{a\} \in \mathrm{Core}(C)$当且仅当存在$k$使得$M[k]$的元素$m_{ij}^k = \{a\}(1 \leq k \leq t)$，且任意$s(1 \leq s \neq k \leq t)$有$M[s]$的元素$m_{ij}^s = \varnothing$.

证明 由定理 1 和推论 1 可得证 □

于是，由定理 4 和定理 1 可得多决策表全局属性核的求解思路如下:

1) 由算法 1 求得全局不一致对象集合U_2，全局一致对象集合$U_1 = U - U_2$，全局不一致对象集合U_2的约简U_2^1;

2) 依据定义 7，在各站点S_i上建立局部决策表DT_i的差别矩阵$M[i]$;

3) 将各局部差别矩阵的行划分为 t 个区间，相应的区间依次为$\left[1, \left[\frac{|U_1|}{t}\right]\right]$，$\left[\left[\frac{|U_1|}{t}\right] + 1, 2\left[\frac{|U_1|}{t}\right]\right]$，$\cdots$，$\left[(t-1)\left[\frac{|U_1|}{t}\right] + 1, |U_1|\right]$;

4) 在站点 S_k，通过求 $m_{ij}\left(i \in \left[(k-1)\left[\frac{|U_1|}{t}\right] + 1, \min\left(k\left[\frac{|U_1|}{t}\right], |U_1|\right)\right], j \in [1, |U_1| + |U_2^1|]\right)$得到仅含单个属性的$m_{ij}$，并将该属性登记到$CC(k)$中;

5) 由在各站点并行求得的各个$CC(k)$得到$\mathrm{Core}(C)$.

依据上述关于垂直分布的多决策表全局属性核的求解思路，可得如下基于垂直分布的多决策表全局属性核求解算法CGAC.

算法 2 CGAC(Computing Global Attribute Core) algorithm.

输入: 有 t 个分布式站点$S_1, \cdots, S_t$，各站点上的局部决策表分别为$DT_i = \langle U, C_i \cup D, V, f\rangle, i = 1, 2, \cdots, t$; 全局决策表为$DT = \langle U, C \cup D, V, f\rangle$，其中$C = \bigcup_{i=1}^{t} C_i$.

输出: 全局属性核$\mathrm{Core}(C)$.

Step 1: 运行算法 1;

Step 2: for $i = 1$ to t do

　　// 各站点并行建立局部差别矩阵$M[i]$

　　create corresponded discernibility matrix $M[i]$ for DT_i;

Step 3: for $k = 1$ to t do

　　// 各站点并行求解$CC(k)$

　　{$CC(k) = \varnothing$;

　　for $i = (k-1)\left[\frac{|U_1|}{t}\right] + 1$ to $\min\left(k\left[\frac{|U_1|}{t}\right], |U_1|\right)$ do

　　　for $j = 1$ to $|U_1| + |U_2^1|$ do

　　　{$m_{ij} = m_{ij}^k$;

　　　if $|m_{ij}| \leq 1$ then 计算m_{ij};

　　　if m_{ij} 为单个属性 then

　　$CC(k) = CC(k) \cup m_{ij}$;

　　}

}

Step 4: $\mathrm{Core}(C) = \bigcup_{k=1}^{t} CC(k)$. // 由定理 4

算法 2 中 Step 1 的时间复杂度为$O(mn\log^n)$; Step 2 的时间复杂度和空间复杂度均为$O(|U_1|(|U_1| + |U_2^1|))$; Step 3 的时间复杂度至多为$O\left(\frac{|U_1|}{t}(|U_1| + |U_2^1|)\right)$. 因此，算法 2 的时间复杂度为$O(mn\log^n) + O(|U_1|(|U_1| + |U_2^1|))$，而将 t 个局部决策表合并后求属性核的时间复杂度至少为$O(|C|n\log^n) + O(|U_1|(|U_1| + |U_2^1|))$，其中$|C| > m$ (当 $t > 1$ 时).

算法 2 的网络通讯量由 Step 1 和 Step 3 计算m_{ij}决定，而 Step 3 计算m_{ij}的网络通讯量可忽略不计，因此算法 2 的网络通讯量与算法 1 的网络通讯量一致.

综上分析可知，算法 2 可有效地降低网络通讯代价，提高算法效率.

3.3 示例说明

例 1 表 1 为局部决策表DT_1和DT_2.

表 1　局部决策表 DT_1 和 DT_2

对象	站点 S_1			站点 S_2		
	a	b	D	c	d	D
x_1	0	0	0	1	2	0
x_2	0	0	1	0	1	1
x_3	0	0	0	0	1	0
x_4	1	1	0	0	0	0
x_5	0	0	1	1	0	1
x_6	0	0	0	2	0	0
x_7	1	2	1	0	0	1
x_8	0	2	0	2	0	0

站点 S_1 上的局部决策表 DT_1 含 8 个对象, 两个条件属性 a 和 b, 一个决策属性 D; 站点 S_2 上的局部决策表 DT_2 含 8 个对象, 两个条件属性 c 和 d, 一个决策属性 D. 根据算法 2 可得运行步骤如下:

Step 1: 运行算法 1. $INCON(1) = \{\langle x_1, x_2\rangle, \langle x_2, x_3\rangle, \langle x_3, x_5\rangle, \langle x_5, x_6\rangle\}$; $INCON(2) = \{\langle x_2, x_3\rangle, \langle x_4, x_7\rangle\}$; $INCON = \{\langle x_2, x_3\rangle\}$. $U_1 = \{x_1, x_4, x_5, x_6, x_7, x_8\}$, $U_2 = \{x_2, x_3\}$, $U_2^1 = \{x_2\}$.

Step 2: 在两个站点建立的局部差别矩阵分别为

$$M[1] = \begin{array}{c|c} & \begin{array}{ccccccc} x_1 & x_4 & x_5 & x_6 & x_7 & x_8 & x_2 \end{array} \\ \begin{array}{c} x_1 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \end{array} & \begin{bmatrix} \varnothing & \varnothing & \varnothing & \varnothing & \{a,b\} & \varnothing & \varnothing \\ \varnothing & \varnothing & \{a,b\} & \varnothing & \{b\} & \varnothing & \{a,b\} \\ \varnothing & \{a,b\} & \varnothing & \varnothing & \varnothing & \{b\} & \varnothing \\ \varnothing & \varnothing & \varnothing & \varnothing & \{a,b\} & \varnothing & \varnothing \\ \{a,b\} & \{b\} & \varnothing & \{a,b\} & \varnothing & \{a\} & \{a,b\} \\ \varnothing & \varnothing & \{b\} & \varnothing & \{a\} & \varnothing & \{b\} \end{bmatrix} \end{array},$$

$$M[2] = \begin{array}{c|c} & \begin{array}{ccccccc} x_1 & x_4 & x_5 & x_6 & x_7 & x_8 & x_2 \end{array} \\ \begin{array}{c} x_1 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \end{array} & \begin{bmatrix} \varnothing & \varnothing & \{d\} & \varnothing & \{c,d\} & \varnothing & \{c,d\} \\ \varnothing & \varnothing & \{c\} & \varnothing & \varnothing & \varnothing & \{d\} \\ \{d\} & \{c\} & \varnothing & \{c\} & \varnothing & \{c\} & \{c,d\} \\ \varnothing & \varnothing & \{c\} & \varnothing & \{c\} & \varnothing & \{c,d\} \\ \{c,d\} & \varnothing & \varnothing & \{c\} & \varnothing & \{c\} & \{d\} \\ \varnothing & \varnothing & \{c\} & \varnothing & \{c\} & \varnothing & \{c,d\} \end{bmatrix} \end{array}.$$

Step 3: 在站点 S_1 可得 $CC[1] = \{b, c, d\}$, 在站点 S_2 可得 $CC[2] = \{b, c\}$.

Step 4: $Core(C) = \{b, c, d\}$.

由例 1 可以看出, 通过局部不一致对象可有效地求出全局不一致对象. 在各局部站点分别建立局部差别矩阵可有效地降低网络通讯量, 提高算法的效率. 此外还可以看出, 差别矩阵 $M[1]$ 可以进一步简化, 由于篇幅限制, 有关局部差别矩阵的优化和进一步的实验验证不再讨论.

4　结　　语

本文引入全局决策表和局部决策表的概念, 提出一种基于垂直分布的多决策表全局属性核求解算法, 该算法可充分利用求得的局部不一致对象和局部差别矩阵, 有效提高垂直分布的全局属性核求解的效率, 为分布式环境下垂直分布的多决策表全局属性核求解提供一种有效的框架. 算法分析结果表明, 本文的算法是有效的, 可行的.

参考文献(References)

[1] Sabine M, David B S. Building Predictors from Vertically Distributed Data[A]. *Proc of the 2004 Conf of the Centre for Advanced Studies on Collaborative Research*[C]. Canada, 2004: 150-162.

[2] Du W L, Zhan Z J. Building Decision Tree Classifier on Private Data [A]. *Proc of the IEEE Int Conf on Privacy, Security and Data Mining* [C]. Maebashi City, 2002: 1-8.

[3] 杨明, 孙志挥, 吉根林. 快速挖掘全局频繁项目集[J]. *计算机研究与发展*, 2003, 40(4): 620-626.
(Yang M, Sun Z H, Ji G L. Fast Mining of Globally Frequent Itemsets [J]. *Computer Research and Development*, 2003, 40(4): 620-626.)

[4] 杨明, 孙志挥, 宋余庆. 快速更新全局频繁项目集[J]. *软件学报*, 2004, 15(8): 1189-1197.
(Yang M, Sun Z H, Song Y Q. Fast Updating of Globally Frequent Itemsets[J]. *J of Software*, 2004, 15(8): 1189-1197.)

[5] Pawlak Z. Rough Sets[J]. *Int J of Information and Computer Science*, 1982, 11(5): 341-356.

[6] Pawlak Z. Rough Set Approach to Multi-attribute Decision Analysis [J]. *European J of Operational Research*, 1994, 72(3): 443-459.

[7] Roman W S, Larry H. Rough Sets as a Front End of Neural-network Texture Classifiers [J]. *Neurocomputing*, 2001, 36(1-4): 85-102.

[8] 蔡忠闽, 管晓宏, 邵萍, 等. 基于粗糙集理论的入侵检测新方法[J]. *计算机学报*, 2003, 26(3): 361-366.
(Cai Z M, Guan X H, Shao P, et al. A New Approach to Intrusion Detection Based on Rough Set Theory[J]. *Chinese J of Computers*, 2003, 26(3): 361-366.)

[9] Jelonek J, Krawiec K, Slowinski R. Rough Set Reduction of Attributes and Their Domains for Neural Networks[J]. *Computational Intelligence*, 1995, 11(2): 339-347.

(下转第 1000 页)

初始总交叉概率为 0.9; 初始总变异概率为 0.1; 初始局部搜索概率为 0.8; 局部搜索次数为 3; 在 SPEA 中外部 Pareto 集大小为 70.

表 2 为用本文提出的算法和 SPEA 算法各运行 10 次的对比结果. 计算结果表明, 与 SPEA 相比, 该方法计算时间减少, 优化解个数相当, 解的性能较好.

表 2 HMOAEA 与 SPEA 优化解性能比较(应用)

目标函数	HMOAEA			SPEA		
	最好	平均	最差	最好	平均	最差
E	0.36	0.67	1.03	0.42	0.73	1.25
N^S	56	52.8	50	56	53.1	51
T^S/s	48.1	49.5	51.3	71.2	75.8	79.3

5 结 语

本文根据棒线材的生产特点, 建立了棒线材批量顺序调度模型, 棒线材的批量调度可简化为单机带各种实际约束的 Flow-shop 调度问题. 研究了改进的基于局部搜索的自适应混合多目标进化算法, 采用变权重组合法获得优化解集, 局部搜索加快算法的收敛速度, 免疫浓度调节保持种群的多样性和解的均匀分布, 自适应调节维持最优的进化操作. 实际现场数据测试结果表明所提出方法的有效性和优越性.

参考文献(References)

[1] Hurink J. An Exponential Neighbourhood for a One-machine Batching Problem[J]. *OR Spektrum*, 1999, 21(4): 461-76.

[2] Jordan C. A Two-phase Genetic Algorithm to Solve Variants of the Batch Sequencing Problem[J]. *Int J of Production Research*, 1998, 36(3): 745-760.

[3] Ovacik IM, Uzsoy R. Rolling Horizon Procedures for Dynamic Parallel Machine Scheduling with Sequence-dependent Setup Times[J]. *Int J of Production Research*, 1995, 33(11): 3173-3192.

[4] 唐立新. 轧钢厂的精轧工序轧制批量调度的优化模型[J]. *东北大学学报*, 1998, 19(6): 624-626.
(Tang L X. Optimal Model of Rolling Lot Scheduling for the Finishing Operation in Rolling Mill[J]. *J of Northeastern University*, 1998, 19(6): 624-626.)

[5] Deb K, Pratap A, Agarwal S, et al. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II[J]. *IEEE Trans on Evolutionary Computation*, 2002, 6(2): 182-197.

[6] Ishibuchi H, Murata T. Multi-objective Genetic Local Search Algorithm[A]. *Proc 3rd IEEE Int Conf Evolutionary Computation*[C]. Nagoya, 1996: 119-124.

[7] Zitzler E, Thiele L. *An Evolutionary Algorithm for Multi-objective Optimization: The Strength Pareto Approach*[R]. Swiss Federal Institute of Technology, 1998.

[8] Alf Kimms. A Genetic Algorithm for Multi-level, Multi-machine Lot Sizing and Scheduling[J]. *Computers and Operations Reaserch*, 1999, 26(8): 829-848.

[9] Talbi. *Meta Heuristics for Multiobjective Combinatorial Optimization: State of the Art*[R]. Lille: University of Lille, 2000.

[10] Hong T P, Wang H S, Chen W C. Simultaneous Applying Multiple Mutation Operators in Genetic Algorithm[J]. *J of Heuristic*, 2000, 6(4): 439-455.

[11] Taillard E. Benchmarks for Basic Scheduling Problems[J]. *European J of Operational Research*, 1993, 64(2): 278-285.

[12] Knowles J D, Corne D W. On Metrics for Comparing Nondominated Sets[A]. *Proc 2002 Congress on Evolutionary Computation*[C]. Honolulu, 2002: 711-716.

(上接第 995 页)

[10] 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 J. A New Discernibility Matrix and the Computation of a Core[J]. *Acta Electronica Sinica Chinese J of Electrics*, 2002, 30(7): 1086-1088.)

[11] 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. Calculation Methods for Core Attributes of Decision Table[J]. *Chinese J of Computers*, 2003, 26(5): 611-615.)

[12] 杨明, 孙志挥. 改进的差别矩阵及其求核方法[J]. *复旦大学学报*, 2004, 43(5): 865-868.
(Yang M, Sun Z H. Improvement of Discernibility Matrix and the Computation of a Core[J]. *J of Fudan University*, 2004, 43(5): 865-868.)

[13] Hu X H, Cercone N. Learning in Relational Databases: A Rough Set Approach[J]. *Computational Intelligence: An International J*, 1995, 11(2): 323-338.