

## 基于时间约束 Petri 网的工作流可调度性分析

冯 林, 姜 浩

(东南大学 计算机科学与工程系, 江苏 南京 210096)

**摘 要:** 引入时间参数控制工作流高效执行是工作流技术研究的重要内容之一。在工作流的 Petri 网模型中引入时间参数, 不仅可以对 workflow 中的活动进行时间约束, 而且能建立相应的工作流时间约束 Petri 网模型。文中在分析该模型的时间约束和可调度性的基础上, 提出了对工作流可调度性分析验证方法及其相应的算法, 最后通过一个实例说明了分析验证的过程。

**关键词:** 工作流; 时间约束 Petri 网; 模型分析; 可调度性

**中图分类号:** TP311.5

**文献标识码:** A

**文章编号:** 1673-629X(2006)11-0034-04

Analysis of Workflow Schedulability Based on Time  
Constraint Petri Nets

FENG Lin, JIANG Hao

(Dept. of Computer Sci. &amp; Tech., Southeast Univ., Nanjing 210096, China)

**Abstract:** It is important for workflow research to introduce time to control execution of workflow effectively. Firstly, time is introduced to workflow models based on Petri nets, and activities in workflow processes are constrained by time. The time constraint Petri net model of workflow is established. Then, based on analysis of time constraint and schedulability of models, the analysis and verification method and corresponding algorithm are proposed. Finally, the analysis and verification process are illustrated by an example.

**Key words:** workflow; time constraint Petri nets; models analysis; schedulability

## 0 引言

在工作流应用日益普及的今天, 工作流管理系统所支持的企业业务过程的正确性、有效性及效率对企业的经营活动是至关重要的。在基于有向图的工作流过程建模方法<sup>[1,2]</sup>中, Petri 网建模方法由于具有严格的数学基础, 因而在工作流模型的分析与验证中得到广泛的应用。而在工作流系统中的时间管理是工作流建模和分析的重要组成部分, 它研究工作流执行的时间维计划, 估计不同活动的执行时间的延迟, 避免活动/过程违反时间约束<sup>[3,4]</sup>, 对时间有界性进行验证<sup>[5]</sup>。文中利用工作流时间约束 Petri 网模型, 提出了一种将局部和全局时间结合起来对工作流中的活动进行时间约束分析和对过程模型的可调度性进行分析的方法, 并给出了相应的验证算法, 还通过实例说明其可行性。

## 1 工作流的时间约束 Petri 网模型

定义1 工作流网<sup>[6]</sup>。

当一个 Petri 网  $PN = (P, T, F, M_0)$  具有如下性质:

\*  $PN$  中有两个特殊的库所: 开始库所  $P_i$  和结束库所  $P_o$ , 开始库所是一个源库所(无输入), 结束库所是一个潭库所(无输出);

\* 对  $\forall x \in P \cup T$ , 存在一条从开始库所到结束库所的路径  $C$ , 使得  $x$  是  $C$  上的一个结点;

\* 网的初始状态为  $M_0 = \{P_i\}$ , 网的终止状态为  $M_e = \{P_o\}$ 。

则称这个 Petri 网为一个工作流网, 记为 WFN。

时间约束 Petri 网是在非时间约束 Petri 网的基础上对变迁的触发引入时间约束而形成的。在 Petri 网中引入时间参数的方式有多种<sup>[7]</sup>, 但大多数的研究工作是采用将时间参数与变迁相关联。而时间参数也有多种形式, 文中采用将每个变迁与一个时间范围相联系的形式。时间约束 Petri 网的状态由网的标识和全局时间来决定。时间 Petri 网由一个五元组定义:  $(P, T, F, M_0, SI)$ , 其中  $(P, T, F, M_0)$  是一个无时间约束 Petri 网,  $SI: T \rightarrow Q \times Q$  是一个由变迁到时间区间的映射函数,  $Q$  是非负有理数集合,  $SI(t) = [a, b]$ , 其中,  $a$  称为相对最早触发时间,  $b$  称为相对最迟触发时间,  $0 \leq a \leq b < \infty$ 。这里的相对是指变迁的实际触发时刻相对于变迁的可触发时刻而言。将映射函数  $SI$  引入定义1的工作流网, 可得到一个时间约束工作流网。

收稿日期: 2006-01-17

作者简介: 冯 林(1982-), 男, 四川遂宁人, 硕士研究生, 研究方向为工作流; 姜 浩, 副教授, 研究方向为工作流应用研究。

定义 2 时间约束 workflow 网(TCWFN)。

时间约束 Petri 网 TCWFN =  $(P, T, F, M_0, SI)$  是一个时间约束 workflow 网, 如果它满足:

\* TCWFN 所对应的无时间约束 Petri 网 WFN =  $(P, T, F, M_0)$  是一个 workflow 网;

\*  $SI$  是变迁到一个时间区间的映射,  $SI: T \rightarrow [a, b], 0 \leq a \leq b < \infty$ 。

这里, WFN =  $(P, T, F, M_0)$  称为与 TCWFN =  $(P, T, F, M_0, SI)$  相对应的无时间约束 Petri 网。在时间约束 workflow 网中, 库所对应过程中的条件, 变迁对应过程中的可执行活动。假定活动 A 对应变迁  $t$ , 则  $t$  的可触发对应活动 A 的开始,  $t$  的相对最早触发时间对应完成活动 A 所需的最少时间,  $t$  的相对最迟触发时间对应于完成活动 A 所需的最长时间。通过这个模型, 可以对工作流过程定义进行时间维上的验证和分析。

## 2 工作流活动的时间约束和可调度性

### 2.1 时间约束

要完成工作流活动的行为分析, 除了将变迁的相对触发时间引入系统之外, 还需要引入全局时间, 并且要将变迁的行为分析与系统的状态结合在一起考虑。为方便描述算法起见, 采用文献[7]中的表示方法。假定变迁  $t$  的可触发时刻为  $\theta$ , 其相对最早触发时间和相对最迟触发时间分别记为  $EFT(t)$ ,  $LFT(t)$ 。令  $I_1 = (a_1, b_1)$ ,  $I_2 = (a_2, b_2)$ ,  $0 \leq a_i \leq b_i < \infty$ , 且  $k$  为一正整数,  $I_1 + I_2 = (a_1 + a_2, b_1 + b_2)$ ,  $I_1 - I_2 = (a_1 - a_2, b_1 - b_2)$ ,  $kI_1 = (ka_1, kb_1)$ 。

在时间约束 workflow 网中, 如果过程实例中的某一活动的发生使得系统在时刻  $\theta$  到达标识  $M$ , 则在标识  $M$  下具有相对触发时间区间  $t(EFT(t), LFT(t))$  的可触发变迁  $t$  必须在  $(\theta + EFT(t), \theta + LFT(t))$  时间区间内触发。用  $ES(M)$  表示在标识  $M$  下可触发的变迁集合。在状态  $M$  下, 可触发变迁  $t$  的可触发实际时间区间不仅被  $SI(t)$  约束, 而且还受其它在  $M$  下可触发变迁  $t' (t' \in ES(M) \wedge t' \neq t)$  的影响。图 1(a) 所示的 Petri 网, 当给定的时间约束如图 1(b) 所示时, 变迁  $t_a$  实际上是不可触发的, 因为在它触发之前,  $t_x$  或  $t_y$  已经触发; 当给定的时间约束如图 1(c) 所示时, 变迁  $t_a$  的可触发实际时间区间由两根虚线所标定。变迁  $t$  在标识  $M$  下的可触发实际时间区间记为  $RT(t) = (RE(t), RL(t))$ , 可由下式计算:

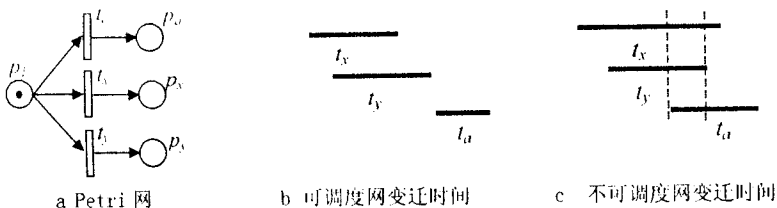


图 1 变迁触发实际可能时间区间分析

$$RT(t) = (RE(t), RL(t)) = (EFT(t), \min\{LFT(t) : t \in ES(M_i)\})$$

变迁的触发将改变系统的状态, 根据工作流过程中完成一个活动所花费的时间对后续活动影响的持续性, 对系统新状态下变迁的时间约束做如下分析: 假如在标识  $M_i$  下某个变迁  $t_i$  的触发使得系统到达状态  $M_{i+1}$ , 变迁  $t_{i+1} \in ES(M)_{i+1}$  的相对触发时间约束的确定可以分为两种情况:

(1) 变迁  $t_{i+1}$  在  $M_i$  下是不可触发的, 而在  $M_{i+1}$  下是可触发的。这时, 变迁的时间约束不受  $t_i$  的影响,  $SI(t_{i+1}) = t(EFT(t_{i+1}), LFT(t_{i+1}))$ 。

(2) 变迁  $t_{i+1}$  在  $M_i$  和  $M_{i+1}$  下都是可触发的。这时, 变迁  $t_{i+1}$  的时间约束受  $t_i$  的可触发实际时间区间的影响, 即:  $SI(t_{i+1}) = (\max\{0, EFT(t_{i+1}) - RL(t_i)\}, LFT(t_{i+1}) - RE(t_i))$ 。

### 2.2 可调度性

定义 3 变迁的可调度。

在一个时间约束 Petri 网中, 当一个可触发的变迁  $t$  在时间区间  $[a, b]$  内的触发不违反时间约束, 则  $t$  在时间区间  $[a, b]$  内被称为是可调度的。

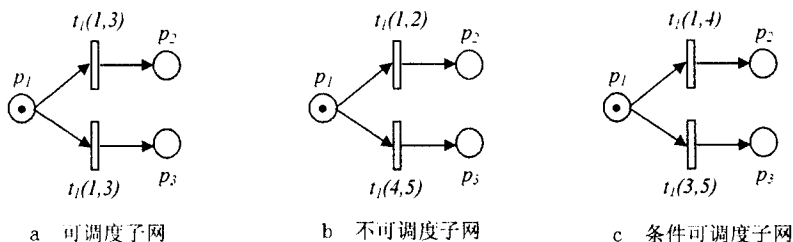


图 2 变迁的可调度性

在图 2 所示的例子中, 在子网(a)中,  $t_1$  和  $t_2$  都是可调度的。在子网(b)中, 只有  $t_1$  是可调度的。在子网(c)中, 时间区间为  $[1, 3]$  时, 只有  $t_1$  可调度; 时间区间为  $[3, 4]$  时,  $t_1$  和  $t_2$  都可调度; 时间区间为  $[4, 5]$  时, 只有  $t_2$  可调度。

特定变迁的可调度性可以通过相对触发时间区间来确定。一般地说, 个别变迁的可调度性依赖于当前标识下所有可触发变迁的时间约束。可以用标识  $M_i$  下变迁的状态类(state classes)<sup>[8]</sup>来描述系统在此状态下所有变迁的时间约束:  $D_i = \{SI(t) : t \in ES(M_i)\}$ , 这实际上是一个在标识  $M_i$  下所有可触发变迁的时间约束的集合, 将其称为标识  $M_i$  下可触发变迁的相对实施域。

定义 4 变迁序列的可调度。

在时间约束 Petri 网中, 当一个变迁序列  $\sigma = t_1, t_2, \dots, t_n$  中所有变迁在给定次序下是可调度的, 即存在标识  $M_0, M_1, \dots, M_n$ , 使得  $(M_0, t_1, M_1, \dots, t_n, M_n)$  在对应的无时间约束 Petri 网中是一个可触发序列, 并且在标识  $M_{i-1}$  下  $t_i (1 \leq i \leq n)$  是可调度的, 则称  $\sigma$  是可调度的, 或者  $\sigma$  是一个调度。

### 3 工作流的可调度性分析

对于分析单个活动可调度性,可以采用相对触发时间域来判断,而分析活动的调度对整个流程时间约束的影响,则利用绝对触发时间域来反映系统从初始状态改变为当前状态的后继状态所有可能的时间约束。状态  $M_i$  下变迁的绝对实施域记为:  $AD_i (1 \leq i \leq n)$ 。变迁  $t$  在  $AD_i$  中的时间区间记为  $AD_i(t) = (AEFT_i(t), ALFT_i(t))$  或者  $t(AEFT_i(t), ALFT_i(t))$ , 其中  $AEFT_i(t)$  和  $ALFT_i(t)$  分别称为  $t$  的绝对最早触发时间和绝对最迟触发时间。

要计算绝对触发时间域,需引入全局时间戳。在一个变迁触发后,可达标识被打上时间戳(变迁的绝对触发时间区间就是系统在该变迁触发后所到达的标识的时间戳)。假定  $TS_i = (AE_i, AL_i)$  是  $M_i$  的时间戳,表示相对于  $M_0$  的标识时刻  $\theta$ ,  $M_i$  在时间区间  $TS_i = (AE_i, AL_i)$  内可达。令初始状态  $M_0$  的时间戳为  $TS_0 = (0, 0)$ ,  $AD_0(t) = \{C(t) : t \in ES(M_0)\}$ 。

为了分析活动流程的一个可能的调度分配  $S$ , 由时间约束 workflow 模型相应的无时间网中得到与  $S$  相对应的一个从开始状态到结束状态的状态变迁序列  $\sigma = M_0, t_1, M_1, t_2, \dots, t_n, M_{n+1}$ 。通过检查状态变迁序列  $\sigma = M_0, t_1, M_1, t_2, \dots, t_n, M_{n+1}$  中每一个变迁  $t_i$  的时间约束特性,可对流程活动的调度  $S$  的可调度性进行分析。下面是具体的分析算法。

#### 算法(变迁序列的可调度性分析)

初始化: 输入状态变迁序列  $\sigma = M_0, t_1, M_1, t_2, \dots, t_n, M_{n+1}$ , 将初始状态  $M_0$  的时间戳定为  $(0, 0)$ , 构造相对触发时间域  $D_0$  和绝对触发时间域  $AD_0$ ,  $D_0 = AD_0 = \{SI(t) : t \in ES(M_0)\}$ , 将状态变迁序列的下标  $i$  置为 0。

第一步: 根据  $M_i$  的相对触发时间域  $D_i$  对变迁  $t_{i+1}$  的可调度性进行判别。如果  $EFT(t_{i+1}) \geq \min\{LFT(t) : t \in ES(M_i)\}$ , 那么,  $t_{i+1}$  在标识  $M_i$  下是不可调度的, 变迁序列  $\sigma = M_0, t_1, M_1, t_2, \dots, t_n, M_{n+1}$  亦为不可调度, 算法结束。否则,  $t_{i+1}$  在标识  $M_i$  下是可调度的。

第二步: 从  $D_i$  构造新的相对触发时间域  $D_{i+1}$ 。

设初值  $D_{i+1} = \emptyset$ , 对于所有在  $M_i$  下是不可触发的, 而在  $M_{i+1}$  下是可触发的变迁  $t$ , 将其静态时间区间  $SI(t)$  加入到  $D_{i+1}$ , 即:  $D_{i+1} = D_{i+1} \cup \{SI(t) : t \in ES(M_i) \wedge t \in ES(M_{i+1})\}$ ; 对于所有在  $M_i$  和  $M_{i+1}$  下都是可触发的变迁  $t$ , 其加入到  $D_{i+1}$  中的时间区间为  $(\max\{0, EFT_{i+1}(t) - RL_i(t)\}, LFT_{i+1}(t) - RE_i(t))$ , 即:  $D_{i+1} = D_{i+1} \cup \{(\max\{0, EFT_{i+1}(t) - RL_i(t)\}, LFT_{i+1}(t) - RE_i(t)) : t \in ES(M_i) \wedge t \in ES(M_{i+1}) \wedge t \neq t_{i+1}\}$ 。

第三步: 从  $AD_i$  构造新的绝对触发时间域  $AD_{i+1}$ 。

设初值  $AD_{i+1} = \emptyset$ , 对于所有在  $M_i$  下是不可触发的, 而在  $M_{i+1}$  下是可触发的变迁  $t$ , 将它的静态时间区间与当前时间戳的和加入绝对触发时间域  $AD_{i+1}$ , 即:

$$AD_{i+1} = AD_{i+1} \cup \{C(t) + TS_{i+1} : t \in ES(M_i) \wedge t \in ES(M_{i+1})\};$$

对于所有在  $M_i$  和  $M_{i+1}$  下都是可触发的变迁  $t$ , 加入到  $AD_{i+1}$  中的时间区间为:

$$AD_{i+1} = AD_{i+1} \cup \{(\max\{AEFT_i(t), AE_{i+1}\}, ALFT_i(t)) : t \in ES(M_i) \wedge t \in ES(M_{i+1}) \wedge t \neq t_{i+1}\}$$

第四步: 计算  $t_{i+1}$  的可调度实际时间区间:

$$(RE_{i+1}(t_{i+1}), RL_{i+1}(t_{i+1})) = (EFT_i(t_{i+1}), \min\{LFT_i(t) : t \in ES(M_i)\})$$

第五步: 计算  $t_{i+1}$  的绝对可调度时间区间, 即状态  $M_{i+1}$  的时间戳:

$$TS_{i+1} = \{AE_{i+1}, AL_{i+1}\} = (AEFT_i(t_{i+1}), \min\{ALFT_i(t) : t \in ES(M_i)\})$$

第六步: 如果状态变迁序列  $\sigma = M_0, t_1, M_1, t_2, \dots, t_n, M_{n+1}$  未分析完,  $i = i + 1$ , 转第一步; 否则, 算法结束。

### 4 一个实例

假设一个生产和经营兼有的企业的简化业务过程如图 3 所示, 其 Petri 网模型如图 4 所示。不难分析出相应的可达图, 如图 5 所示。

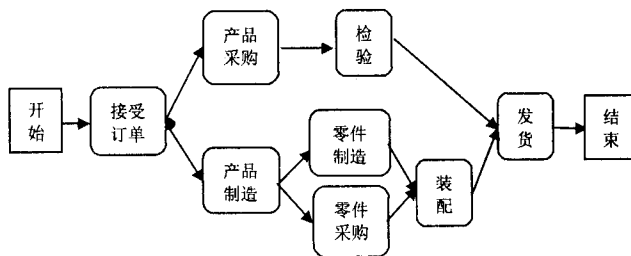


图 3 一个简化的业务过程

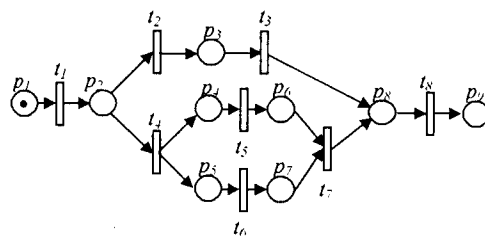


图 4 对应的时间约束工作流网

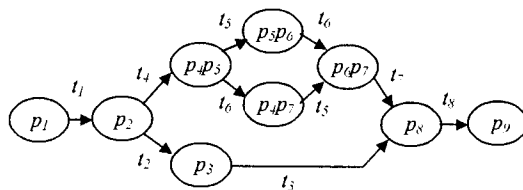


图 5 Petri 网的可达图

假定图 3 所示的工作流网的时间约束由表 1 给定。对于变迁序列  $\delta = t_1, t_4, t_5, t_6, t_7, t_8$ , 通过上述算法分析, 分析结果如表 2 所示, 该序列是可调度的, 该调度序列总的执行时间区间为  $(10, 20)$ 。其中, 任一子序列的执行时间区间可以很容易地推出, 如子序列  $\delta' = t_5, t_6, t_7$  的执

行时间为  $(8, 16) - (4, 8) = (4, 8)$ 。如果对变迁序列  $\delta = t_1, t_4, t_6, t_5, t_7, t_8$  进行同样的分析, 可以发现, 由于变迁  $t_6$  是不可调度的, 因此, 该序列成为不可调度。

表 1 变迁的时间约束

变迁	EFT	LFT
$t_1$	1	2
$t_2$	1	2
$t_4$	2	4
$t_5$	1	2
$t_6$	5	7
$t_7$	1	2
$t_8$	2	4

表 2 可调度性分析过程

序号	时间戳	状态 $M$	$D_i$	$AD_i$	$RT$	$TS$
0	(0, 0)	$\{P_0\}$	$\{t_1(1, 2)\}$	$\{t_1(1, 2)\}$	(1, 2)	$t_1 : (1, 2)$
1	(1, 2)	$\{P_2\}$	$\{t_2(1, 2), t_4(2, 4)\}$	$\{t_2(2, 4), t_4(3, 6)\}$	(2, 2)	$t_4 : (3, 4)$
2	(3, 6)	$\{P_4, P_5\}$	$\{t_5(1, 2), t_6(5, 7)\}$	$\{t_5(4, 8), t_6(8, 13)\}$	(1, 2)	$t_5 : (4, 8)$
3	(4, 8)	$\{P_5, P_6\}$	$\{t_6(3, 6)\}$	$\{t_6(7, 14)\}$	(3, 6)	$t_6 : (7, 14)$
4	(7, 14)	$\{P_6, P_7\}$	$\{t_7(1, 2)\}$	$\{t_7(8, 16)\}$	(1, 2)	$t_7 : (8, 16)$
5	(8, 16)	$\{P_8\}$	$\{t_8(2, 4)\}$	$\{t_8(10, 20)\}$	(2, 4)	$t_8 : (10, 20)$

## 5 结 论

将时间参数引入工作流模型中并对模型进行时间维上的分析是工作流技术的重要研究内容。文中采用工作流的时间约束 Petri 网模型, 提出了工作流过程可调度性

(上接第 33 页)

但利用文中所提出的方法, 首先利用公式(2) 找到相对核  $Core(C) = \{x_2, x_3, x_5\}$ , 依次文中的算法, 上面的区分矩阵可化简如表 3 所示。

表 3 简化后的区分矩阵

1	2	3	4	5	6	7	8	9	10
1									
2	0								
3		0							
4	0		0						
5	0			0					
6		0		0	$x_1 \times 4$				
7		0		0		0			
8	0		0				0	$x_1 \times 4$	
9	0		0				0	$x_1 \times 4$	
10		0		0				$x_1 \times 4$	0

则得到的析取逻辑表达式得到极大简化, 为:  $x_1 \vee x_4$ 。

最后把所有的析取逻辑表达式进行合取运算则仍为:  $x_1 \vee x_4$ 。

可以很容易地得到此决策表的属性约简为:  $\{x_1, x_2, x_3, x_5\}$  和  $\{x_4, x_2, x_3, x_5\}$ 。

## 4 结束语

基于 Rough Set 的机器学习理论是数据挖掘的一个

分析的方法和具体的实现算法, 将可调度性分析与活动执行时间的估计结合起来, 为工作流的高效执行提供了理论和方法上的支持。

## 参考文献:

- [1] 罗海滨, 范玉顺, 吴 澄. 工作流技术综述[J]. 软件学报, 2000, 11(7): 899 - 907.
- [2] 潘启澍, 姜 兵. 基于 Petri 网的工作流建模技术及应用[J]. 清华大学学报, 2000, 4(9): 86 - 89.
- [3] 李慧芳, 范玉顺. 工作流系统时间管理[J]. 软件学报, 2002, 13(4): 1552 - 1557.
- [4] Eder J, Panagos E. Time management in workflow systems [C]//Proc. International conference in business information systems. Heidelberg, Berlin: Springer - Verlag, 1999: 265 - 280.
- [5] 李建强, 范玉顺. 工作流模型时间有界性验证与分析研究[J]. 计算机集成制造系统, 2002, 8(10): 770 - 775.
- [6] Ver Der Aalst W M P. Verification of workflow task structures [J]. Information systems, 2000, 25(1): 43 - 69.
- [7] Bowden E D J. A brief survey and synthesis of the roles of time in Petri nets[J]. Mathematical and computer modeling, 2000, 31(10 - 12): 55 - 68.
- [8] Tsai J J P, Yang S J. Timing constraint Petri nets and their application to schedulability analysis of real - time system specifications[J]. Transactions on the software engineering, 1995, 21(1): 32 - 49.

很重要的方法。简化和改善 Rough Set 的属性约简和决策规则约简, 是提高数据挖掘能力和效率的重要途径。

文中在区分矩阵的基础上, 提出了一种新的改进的计算信息系统属性约简的方法。该方法把数学逻辑运算运用于区分矩阵中相对核属性以外的其它属性组合, 并以求核为起点化简了区分矩阵, 从而使属性约简计算量大幅度减小, 快速得到给定要求下的属性约简。

## 参考文献:

- [1] Dutsch I. A Logic for Rough Sets[J]. Theoretical Computer Science(B), 1997, 179: 427 - 436.
- [2] 韩祯祥, 张 琦, 文福拴. 粗糙集理论及其应用综述[J]. 控制理论与应用, 1999, 16(2): 153 - 156.
- [3] 张文修, 吴伟志, 梁吉业, 等. 粗糙集理论与方法[M]. 北京: 科学出版社, 2001.
- [4] 王 珏, 王 任, 苗夺谦, 等. 基于 Rough Set 理论的数据浓缩[J]. 计算机学报, 1998, 21(5): 393 - 399.
- [5] Wang Hui, Dutsch I, Gediga G. Classificatory filtering in decision systems[J]. International Journal of Approximate Reasoning, 2000, 23: 111 - 136.
- [6] 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.