

## 基于相邻事件概率统计的流程挖掘方法

石美红,曹开端,陈亮,王泉峰

(西安工程大学 计算机科学学院, 西安 710048)

(meihong\_shi@263.net; kaiduan611@yahoo.com.cn)

**摘要:**为了提高流程挖掘的准确性和抗噪性,针对目前流程挖掘的基本结构有限、抗噪能力弱、计算耗时长等问题,提出了一种基于相邻事件概率统计的流程挖掘方法。该方法基于挖掘规则,仅需做一次日志遍历和矩阵的简单运算,就可生成挖掘的流程模型。与 $\alpha$ 算法和启发式算法的实验验证结果表明,该算法不仅能够挖掘顺序、选择、并行、短循环、递归等流程基本结构,而且具有计算复杂度低、抗噪能力强等优势。

**关键词:**流程挖掘;相邻事件;概率统计;噪声

**中图分类号:** TP311.13 **文献标志码:** A

### Method of process pattern mining based on probability statistics of adjacent event

SHI Mei-hong, CAO Kai-duan, CHEN Liang, WANG Quan-feng

(School of Computer Science, Xi'an Polytechnic University, Xi'an Shaanxi 710048, China)

**Abstract:** To mine accurately process patterns and improve anti-noise ability, a new method of process pattern mining based on probability statistics of adjacent event was proposed to generate automatically process patterns mining based on rules through only once traversing logs and simply calculating matrix. Comparing with  $\alpha$ -algorithm and heuristic algorithm, the results from analyzing mining performance and verifying experiment show that it can mine pattern structures of sequence, choice, paralleling, recursion and short loop, and has advantages of low complexity and good anti-noise ability.

**Key words:** process mining; adjacent event; probability statistics; noise

## 0 引言

目前,支持业务流程建模的工作流技术在办公自动化、工业制造等领域得到了广泛应用。工作流技术应用的核心问题是构造符合业务流程需求的流程模型。由于流程模型随着需求的变化而时常发生变更,其构建需要深厚的领域知识支持,因此,业务流程建模不仅繁琐、费时,而且所建的模型往往与实际业务流程有偏差。事实上,工作流系统应用之前,业务流程信息已经被其他系统以事件的形式记录在日志中。通过分析日志数据,自动生成流程模型的过程便是工作流挖掘,也称为流程挖掘<sup>[1]</sup>。流程挖掘不仅可以监控到业务流程模型在实际运行过程的变动,能够构建具有客观性、准确性的流程模型,而且也为了提高模型构建的效率、优化业务流程的模型、促进工作流技术朝着柔性化方向的发展提供重要的支持。

文献[2]最早提出了基于有向图的工作流挖掘模型,该方法只能发现流程中的顺序和循环结构,但活动间的并行关系难以确定。文献[3]提出了基于工作流网的 $\alpha$ 流程挖掘算法,该算法通过日志中活动之间的时序关系挖掘流程结构,其特点是方法简单,计算复杂度低,但处理噪声能力弱,且对于循环、递归等流程结构模型的挖掘能力有限。文献[4]提

出了一种启发式的挖掘方法,通过启发式规则和频率/依赖表从事件日志中推导出工作流模型,该方法能够从不完整的日志中挖掘出准确流程模型,然而对于短循环和递归结构却需要增加复杂的规则来处理,且算法复杂度较高。文献[5]提出一种基于最大依赖性的启发式规则的流程挖掘思想,但因其判定条件来自于启发式方法,所以算法复杂,且无法直接判定短循环和递归结构。

针对已有方法存在的问题,本文提出了一种基于相邻事件概率统计的流程挖掘方法。本方法根据日志中相邻事件概率越大其对应活动的依赖关系越强的思想,通过比较相邻事件发生的概率统计值来确定基本流程模型,通过挖掘规则,以此建立可含有顺序、选择、并行、循环、短循环和递归等基本结构的流程模型。该方法具有一定的抗噪能力且计算复杂度低。

## 1 相关定义

**定义1** 流程日志,流程路径。设 $W$ 为流程日志, $T$ 是流程日志中的活动集合,即 $T = \{t_1, t_2, \dots, t_i, \dots, t_n\}$ ,  $i = \{1, 2, \dots, n\}$ ,  $\sigma$ 是一事件序列所产生的一条流程路径,那么, $\sigma \in T^*$ ,  $W \in \rho(T^*)$ , 其中, $T^*$ 表示 $T$ 上任意长度的子集序列, $\rho(T^*)$ 是 $T$ 的所有子集序列的集合。

收稿日期:2010-09-20;修回日期:2010-11-22。

基金项目:国家科技支撑计划项目(2006BAF01A44);陕西省自然科学基金资助项目(SJ08-ZT14)。

作者简介:石美红(1956-),女,江苏仪征人,教授,主要研究方向:图像处理、模式识别、智能信息处理;曹开端(1985-),男,河南开封人,硕士研究生,主要研究方向:流程挖掘、系统集成;陈亮(1977-),男,湖南怀化人,副教授,博士,主要研究方向:过程管理、数据挖掘;王泉峰(1985-),男,河南洛阳人,硕士研究生,主要研究方向:协同设计、知识管理。

**定义2** 前驱活动,后继活动,依赖关系。设  $t_i, t_j \in T$ , 若在流程日志中存在一个由  $t_i$  到  $t_j$  的流向, 且  $t_i$  到  $t_j$  之间没有其他活动, 则  $t_i$  是  $t_j$  的前驱活动,  $t_j$  是  $t_i$  的后继活动,  $t_i$  与  $t_j$  之间存在依赖关系 ( $t_j$  依赖于  $t_i$ ), 记作  $t_i \rightarrow t_j$ 。

**定义3** 短循环。设  $t_i, t_j \in T$ , 若  $t_i \rightarrow t_j$  且  $t_j \rightarrow t_i$ , 则  $t_i, t_j$  构成一个短循环结构, 记作  $t_i \leftrightarrow t_j$ 。

**定义4** 递归结构。设  $t \in T$ , 若  $t \rightarrow t$ , 则  $t$  构成一个短递归结构。

**定义5** 工作流网 WF-net<sup>[6]</sup>。设工作流网是一个三元组  $WF-net = \langle P, T, F \rangle$ , 其中,  $P$  是库所的集合,  $T$  是变迁 (即活动) 的集合,  $F$  是库所与变迁关系的集合, 当且仅当满足:

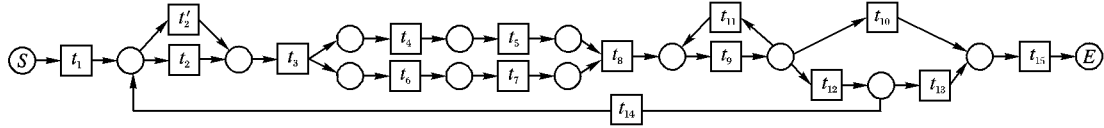


图1 一个工作流网描述的流程模型

**定义6** 相邻事件关系。如果在流程日志  $W$  中存在一条流程路径  $\sigma = \dots t_{j+1} \dots t_k$ ;  $j, k \in \{1, 2, \dots, n\}$ , 满足  $\sigma \subseteq W$ , 则  $t_j$  是  $t_{j+1}$  的前驱事件,  $t_{j+1}$  是  $t_j$  的后继事件,  $t_j$  与  $t_{j+1}$  之间存在相邻事件关系, 记作  $t_j >_w t_{j+1}$ 。

**定义7** 相邻事件的活动依赖关系。设  $a, b, t_i \in T$  是  $W$  中记录的流程路径上所产生的活动事件,  $\forall t_i \in T, i \in \{1, 2, \dots, n\}$  且  $t_i \neq a, t_i \neq b$ , 如果  $\text{Count}(a >_w b) \geq \text{Count}(a >_w t_i)$  或  $\text{Count}(a >_w b) \geq \text{Count}(t_i >_w b)$ , 那么,  $a$  和  $b$  间存在相邻事件的活动依赖关系, 即  $a$  是  $b$  的相邻事件的前驱活动,  $b$  是  $a$  的相邻事件的后继活动。其中,  $\text{Count}(x)$  是相邻事件关系  $x$  发生的次数。

**定义8** 噪声对每个活动的影响度。噪声是指因受外界或不确定因素的干扰而造成日志含有错误信息或不完整信息, 设噪声对每个活动的影响度  $\delta = 1 + \text{Round}(N \times m/n)$ <sup>[4]</sup>, 其中,  $n$  为流程日志中活动总数;  $m$  是流程日志中流程路径数量,  $N$  为占流程日志中噪声事件的含量。

## 2 基于相邻事件概率统计的流程挖掘方法

### 2.1 挖掘规则

假设  $W$  为流程日志,  $T = \{t_1, t_2, \dots, t_i, \dots, t_j, \dots\}$ ,  $i, j \in \{1, 2, \dots, n\}$  是  $W$  中记录的活动集合,  $\forall a, b, t_i, t_j \in T$  是流程路径上所产生的事件活动, 那么, 判断各活动之间的依赖关系和流程结构的规则如下。

**规则1** 如果对于  $\forall t_i \in T$ , 都有  $\text{Count}(a >_w b) \geq \text{Count}(t_i >_w b) > \delta$  或  $\forall t_j \in T$ , 都有  $\text{Count}(a >_w b) \geq \text{Count}(a >_w t_j) > \delta$ , 则,  $a, b$  间存在活动依赖关系, 即  $a \rightarrow b$ 。

**规则2** 如果存在  $a \rightarrow t_i, a \rightarrow t_j$  的相邻事件活动依赖关系, 且存在  $\text{Count}(t_i >_w t_j) > \delta$  或  $\text{Count}(t_j >_w t_i) > \delta$ , 那么,  $t_i$  和  $t_j$  之间构成并行结构; 否则为选择结构。

**规则3** 如果存在  $a \rightarrow b$  且  $b \rightarrow a$  的相邻事件的活动依赖关系, 那么,  $a, b$  之间构成了一个短循环结构。

**规则4** 若存在  $a \rightarrow a$  的相邻事件的活动依赖关系, 则  $a$  构成了一个递归结构。

### 2.2 算法思想与算法描述

本文算法的基本思想是: 根据日志中相邻事件概率越大其对应活动的依赖关系越强的思想, 首先通过计算相邻事件

- 1) 对象创建。  $P$  包含一个输入库所  $p_i$ , 使得  $\cdot p_i = \emptyset$ 。
- 2) 对象完成。  $P$  包含一个输出库所  $p_o$ , 使得  $p_o \cdot = \emptyset$ 。
- 3) 强连接性。如果在 WF-net 中加入一个新变迁  $t$ , 使  $t$  连接库所  $p_o$  和  $p_i$ , 即  $t \cdot = \{p_i\}$ ,  $\cdot t = \{p_o\}$ , 这时所得到的 WF-net 是强连接的。

图1是一个由顺序、并行、选择和循环等基本结构组成的工作流网模型。其中,  $S$  为输入库所;  $E$  为输出库所;  $t_2$  与  $t_2'$  执行相同事件, 产生相同的日志记录  $t_2$ , 它们构成了一个递归结构;  $t_9$  与  $t_{11}$  构成了一个短循环结构。在日志中记录的一条流程路径为:  $t_1, t_2, t_2, t_3, t_6, t_4, t_5, t_7, t_8, t_9, t_{11}, t_9, t_{11}, t_9, t_{12}, t_{14}, t_2, t_3, t_6, t_4, t_5, t_7, t_8, t_9, t_{10}, t_{15}$ 。

关系, 构造一个基本流程依赖关系矩阵, 然后, 在此基础上, 基于挖掘规则确定流程的基本结构, 进而得到完整、一致的流程模型。其具体的实现步骤: 1) 建立相邻事件关系矩阵。遍历日志中记录的所有流程路径, 统计相邻事件关系发生的次数, 构件一个相邻事件关系矩阵。2) 确定基本流程活动依赖关系。依据规则1和构建的相邻事件关系矩阵, 产生基本流程活动依赖关系矩阵。3) 生成流程模型。基于挖掘规则, 确定基本流程活动依赖关系矩阵中的流程各种结构, 最后自动生成流程模型图。

#### 2.2.1 统计相邻事件关系

假设  $T = \{t_1, t_2, t_3, \dots, t_n\}$  为  $W$  中记录的活动集合, 那么, 定义一个大小为  $n \times n$  的二维矩阵  $D$  用于统计各相邻事件关系发生的次数, 则矩阵中的值  $D_{ij} = \text{Count}(t_i >_w t_j)$ 。如果  $W$  中记录了  $m$  条流程路径, 那么, 建立相邻事件关系矩阵的具体操作如下:

第1步 初始化矩阵  $D = n \times n$  为0。

第2步 遍历日志中记录的所有流程路径, 对于每条流程路径  $\sigma_i$ , 进行如下操作:

1) 将属于流程路径  $\sigma_i$  而不属于已统计的活动集  $T$  的活动集  $T^*$  加入到  $T$  中, 扩充  $D$  矩阵  $n \times n = (n + |T^*|) \times (n + |T^*|)$ , 使其包含  $T^*$ , 并对扩充的矩阵元素初始化。

2) 对于流程路径  $\sigma_i$  中每一个满足定义6的相邻事件关系, 其对应的  $D_{ij}$  值加1, 表示在该流程路径中出现过一次  $t_i >_w t_j$ 。

重复第2步, 最后将得到一个包含日志中所有存在相邻事件关系的矩阵  $D = n \times n$ 。对  $W$  日志统计的相邻事件关系矩阵如下:

| $t_1$    | $t_2$    | $t_3$    | $t_4$    | $t_5$    | $t_6$    | $t_7$    | $\dots$  | $t_{15}$ |          |
|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| 0        | 100      | 2        | 0        | 0        | 0        | 0        | $\dots$  | 0        | $t_1$    |
| 0        | 338      | 137      | 0        | 6        | 0        | 0        | $\dots$  | 0        | $t_2$    |
| 0        | 0        | 3        | 69       | 0        | 68       | 0        | $\dots$  | 0        | $t_3$    |
| 0        | 1        | 0        | 0        | 85       | 33       | 19       | $\dots$  | 0        | $t_4$    |
| 0        | 0        | 0        | 0        | 0        | 36       | 35       | $\dots$  | 0        | $t_5$    |
| 0        | 0        | 0        | 37       | 17       | 0        | 83       | $\dots$  | 0        | $t_6$    |
| 0        | 0        | 0        | 31       | 35       | 0        | 0        | $\dots$  | 0        | $t_7$    |
| $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ |
| 0        | 0        | 0        | 0        | 0        | 0        | 0        | $\dots$  | 0        | $t_{15}$ |

### 2.2.2 确定活动依赖关系

定义一个与  $D$  同等大小的矩阵  $DT$ , 并初始化为 0, 那么, 依据规则 1, 确定基本流程活动依赖关系矩阵的具体操作如下:

第 1 步 计算最大后继关系。对于  $D$  矩阵中的每一行  $i$ , 取非主对角线上大于  $\delta$  的  $D_{ij} (1 \leq i, j \leq n, i \neq j)$  值, 如果  $D_{ij} = \max\{D_{ik}\}, 1 \leq k \leq n$ , 则  $DT_{ij} = D_{ij}$ 。

第 2 步 计算最大前驱关系。对于  $D$  矩阵中的每一列  $j$ , 取非主对角线上大于  $\delta$  的  $D_{ij} (1 \leq i, j \leq n, i \neq j)$  值, 如果  $D_{ij} = \max\{D_{kj}\}, 1 \leq k \leq n$ , 则  $DT_{ij} = D_{ij}$ , 然后, 统计  $D$  矩阵中主对角线上  $D_{ij} (i = j) > \delta$  的元素, 将其记录到  $DT$  中, 即  $DT_{ij} = D_{ij}$ 。

最后将得到一个  $DT = n \times n$  基本流程活动依赖关系矩阵。对上述相邻事件关系矩阵的计算的结果如下。

| $t_1$    | $t_2$    | $t_3$    | $t_4$    | $t_5$    | $t_6$    | $t_7$    | $\dots$  | $t_{15}$ |
|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| 0        | 100      | 0        | 0        | 0        | 0        | 0        | $\dots$  | 0        |
| 0        | 338      | 137      | 0        | 6        | 0        | 0        | $\dots$  | 0        |
| 0        | 0        | 0        | 69       | 0        | 68       | 0        | $\dots$  | 0        |
| 0        | 1        | 0        | 0        | 85       | 0        | 0        | $\dots$  | 0        |
| 0        | 0        | 0        | 0        | 0        | 0        | 0        | $\dots$  | 0        |
| 0        | 0        | 0        | 0        | 0        | 0        | 83       | $\dots$  | 0        |
| 0        | 0        | 0        | 0        | 0        | 0        | 0        | $\dots$  | 0        |
| $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\ddots$ | $\vdots$ |
| 0        | 0        | 0        | 0        | 0        | 0        | 0        | $\dots$  | 0        |

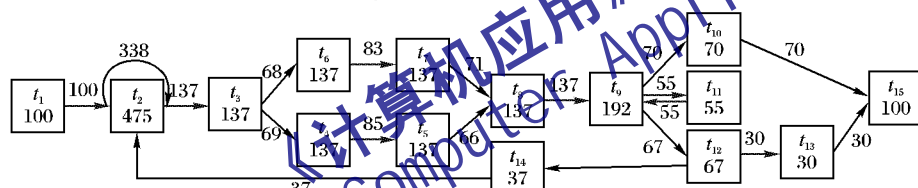


图2 生成的有向图表示的基本流程模型

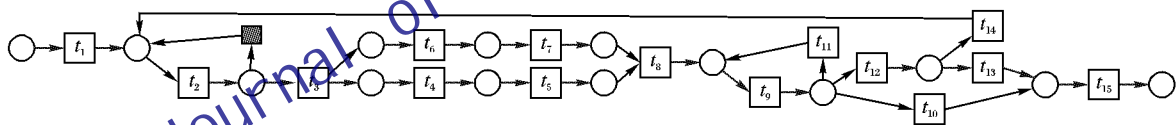


图3 由图2挖掘的流程模型

### 2.3 挖掘方法改进

本方法不能挖掘出图 4 所示的复杂流程结构。若图 4 中  $t_3$  挖掘出的前序活动是  $t_1$ , 而  $t_2$  的后继活动是  $t_4$  时, 将会丢失  $t_2$  到  $t_3$  活动间的依赖关系, 导致挖掘失败。对于这种情况, 可以观察到, 如果  $t_3$  有两个前驱活动  $t_1$  和  $t_2$ , 那么在流程日志  $W$  中就存在  $t_1 >_w t_3$  且  $t_2 >_w t_3$  相邻事件的活动依赖关系, 同理也存在  $t_2 >_w t_3$  且  $t_2 >_w t_4$  相邻事件的活动依赖关系。因此, 可以改进规则 1。

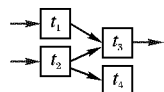


图4 复杂流程结构

规则 5 如果对于  $\forall t_i \in T$ , 都有  $\text{Count}(a >_w b) \geq \max\{\text{Count}(t_i >_w b)\} \times f > \delta$  或  $\forall t_j \in T$ , 都有  $\text{Count}(a >_w b) \geq \max\{\text{Count}(a >_w t_j)\} \times f > \delta$ , 则  $a, b$  间存在活动依赖关系, 即  $a \rightarrow b$ , 其中,  $f$  为复杂度系数, 取值  $(0, 1]$ 。

### 3 算法性能对比分析

本挖掘算法已作为开源流程挖掘框架 ProM<sup>[7]</sup> 的插件

### 2.2.3 生成流程模型

由基本流程活动依赖关系矩阵确定流程的基本结构。其具体操作如下:

第 1 步 生成基本流程模型: 如果  $DT$  矩阵中  $DT_{ij}$  元素大于 0, 表明  $t_i$  与  $t_j$  间存在相邻事件的活动依赖关系  $t_i \rightarrow t_j$ , 即表示它们之间有一条连接边。将  $DT$  矩阵中存在活动依赖关系的各活动连接起来, 便得到了基本的流程。图 2 是由上述活动依赖关系矩阵生成的基本流程模型。

第 2 步 依据规则 2, 对于  $DT$  矩阵中的每一行  $i$ , 若存在两列  $j$  和  $k$ , 使得  $DT_{ij} > 0, DT_{ik} > 0, i \neq j \neq k$ , 且矩阵  $D$  中  $D_{jk} > \delta$  或  $D_{kj} > \delta$  成立, 则,  $t_j$  与  $t_k$  间存在相邻事件的并行结构, 否则标识为分支结构。同理, 对于  $DT$  矩阵中的每一列  $j$ , 判断每两行间是否存在并行或分支结构。

第 3 步 依据规则 3, 判断  $DT$  矩阵中关于主对角线对称的元素值  $DT_{ij}, DT_{ji}$ , 如果  $DT_{ij} > 0$  且  $DT_{ji} > 0$ , 则表明  $t_i$  与  $t_j$  间存在短循环结构。

第 4 步 依据规则 4, 判断  $DT$  矩阵主对角线上  $DT_{ij} (i = j) > 0$ , 如果  $DT_{ij} > 0$ , 表明  $t_i$  与  $t_j$  间存在递归结构。

基于挖掘规则 2~4, 确定基本流程模型中的基本结构, 最后自动生成流程模型图。图 3 是基于挖掘规则由图 2 生成的流程模型。

StatisticMiner 加以实现。它以标准的 MXML 格式的日志作为输入, 其结果返回由有向图表示的基本流程模型和用 WF-net 表示的流程模型 (如图 5 所示)。

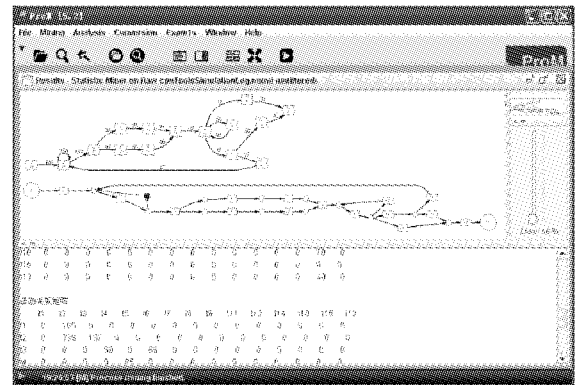


图5 流程挖掘算法实现的运行结果

在流程挖掘的结构正确性、抗噪和挖掘耗时等性能指标上, 与  $\alpha$  算法、启发式算法比较, 本文方法具有一定的优势。在流程结构挖掘方面, 它不仅可以挖掘出顺序、选择、并行、循环等基本结构, 而且可以挖掘出短循环和递归结构; 在抗噪能力方面, 基于相邻事件概率统计的最大值方法, 其本身就具有

屏蔽噪声的作用,在挖掘过程中又考虑了噪声因子,使得该方法具有更强的抗噪能力;在挖掘耗时方面,设流程日志中流程路径数为  $m$ , 活动数为  $n$ , 本方法的时间复杂度约为  $O(mn^2) + O(n^2) \approx O(mn^2)$ ,  $\alpha$  算法的计算耗时在基本关系的查找和流程模型的构建基础上,时间复杂度为  $O(mn^2) + O(n^2) \approx O(mn^2)$ ; 启发式算法的时间复杂度为  $O(mn^3) + O(n^2) + O(n^2) \approx O(mn^3)$ 。本文进行了多组对比实验验证和一致性测试<sup>[8]</sup>,图6(含有1000条路径15个活动18666个事件,有短循环和递归,无噪声)和图7(含有73条路径11个活动440个事件,有短循环和递归,有噪声,含噪20%)给出了

其中两组实验的结果。测试平台为一台 Intel 双核 Pentium D 3.00 GHz、内存 2.00 GB DDR II, 硬盘 150 GB/7 200 转/8 MB 的微型计算机,实验数据来自 ProM 开发小组提供的日志和采用 CPN Tools<sup>[9]</sup> 模拟的日志。

从挖掘的实验结果看,在含有少量噪声( $\leq 20\%$ )的情况下,与其他算法相比,本方法具有较高的合理性和结构的合适性,尤其对含有短循环和递归结构的流程模型挖掘,其结果明显优于其他算法。在抗噪能力方面,本文方法与启发式算法相当,但优于  $\alpha$  算法。在耗时上,本文方法与  $\alpha$  算法相当,但明显优于启发式算法。

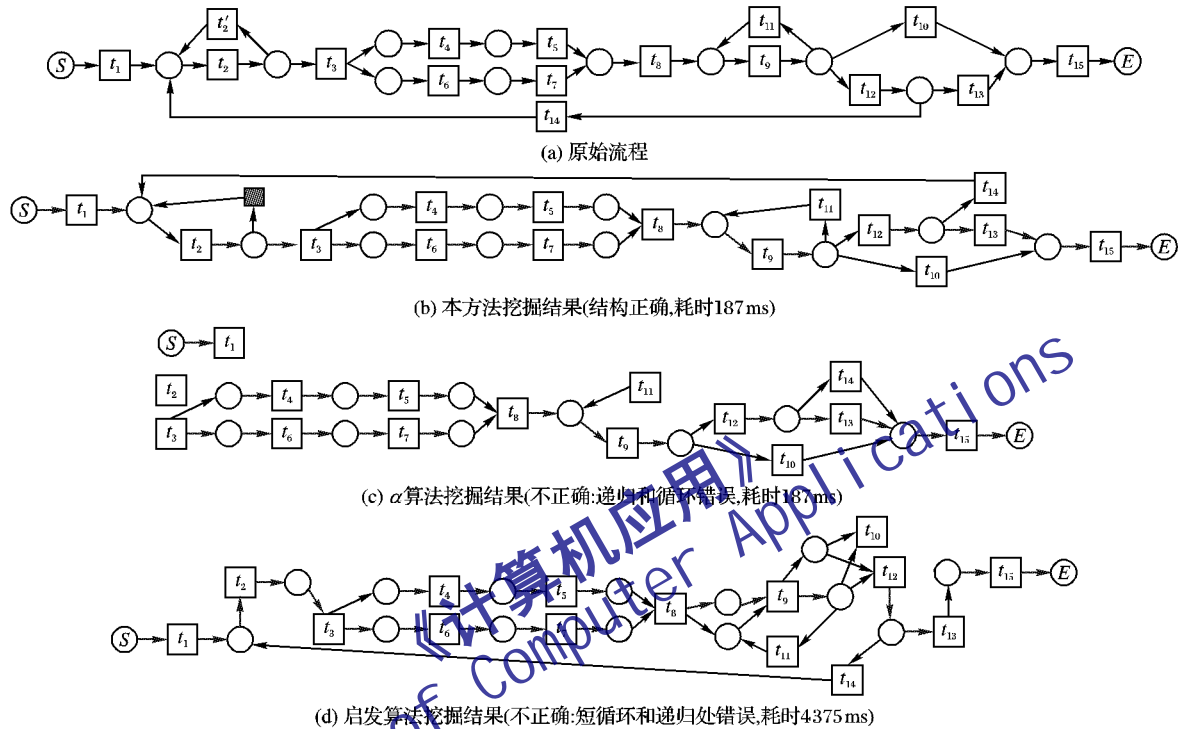


图6 流程日志 cpnLog4.xml 的挖掘结果

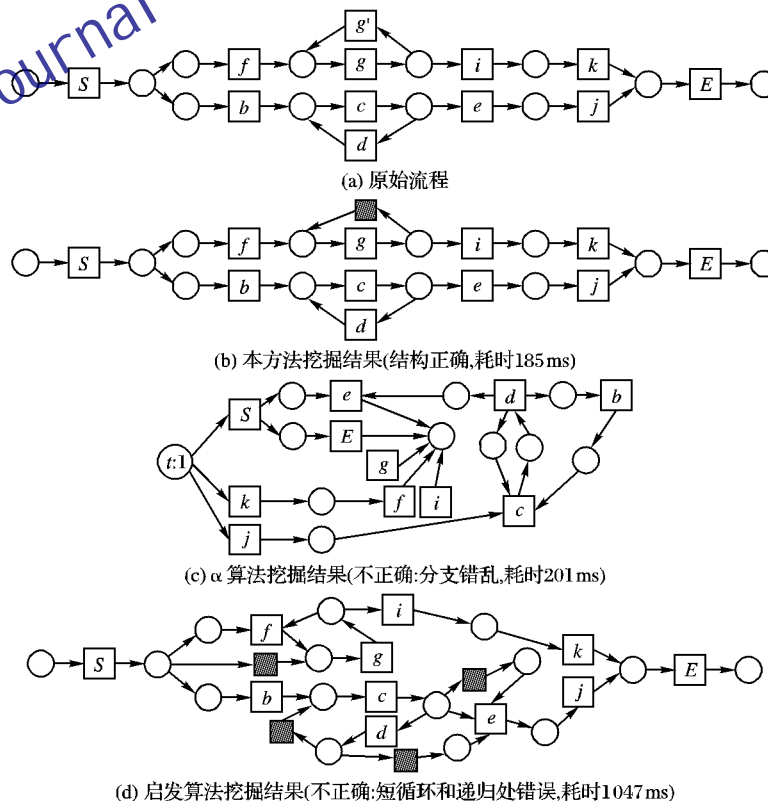


图7 流程日志 a12f0n20.xml 的挖掘结果

(下转第 1394 页)



度的规则的精简表示形式。

经过上面的分析和处理,选择爱立信的四类网元设备的告警进行关联规则挖掘,挖掘结果如表2所示。交叉支持度阈值为0.01,最小置信度为0.8。

目前已确认的爱立信告警关联规则共36条,挖掘导出的关联规则覆盖了其中30条,覆盖率达83.3%,达到了较好的挖掘效果。另外,根据已有的告警数据库,挖掘出了大量有价值的潜在告警关联规则。

表2 爱立信关联规则挖掘结果

| 网元类别       | 类别名称      | 告警数     | 导出规则数 |
|------------|-----------|---------|-------|
| STP        | 信令转接点     | 165 989 | 107   |
| MSC-Server | 移动交换中心    | 127 066 | 285   |
| TMSC       | 汇接局移动交换中心 | 40 249  | 40    |
| MWG        | 媒体网关      | 23 235  | 62    |

#### 4 结语

本文提出并实现了基于递归回溯的无项头表的频繁模式挖掘算法 NHTFPG,解决了 FP-Growth 算法对频繁模式树的公共路径造成重复遍历的问题。理论分析和实验结果表明 NHTFPG 在时间性能和空间性能上均优于 FP-Growth 算法。在通信网络告警关联规则挖掘中的应用表明,NHTFPG 可快速挖掘出符合要求的告警关联规则。尽管加上交叉支持模式约束和置信度约束,使得生成的关联规则大大精简,但仍存在较多的冗余甚至无效的规则。因此,深入分析通信网络及其告警的特征,将挖掘算法与专业领域的知识相结合,使得生成的规则更精简,更有效,还有很多工作要做。此外,将 NHTFPG 生成的有效告警关联规则和专家系统相结合,以实现有效的告警处理和快速故障精确定位是进一步研究的方向。

(上接第1381页)

#### 4 结语

针对目前已有方法存在着流程挖掘的基本结构有限、抗噪能力弱、计算耗时长等问题,本文提出了一种基于相邻事件概率统计的流程挖掘方法。首先阐述了挖掘规则,然后描述了算法的思想和方法的实现步骤,最后通过流程结构正确性、抗噪和挖掘耗时等性能指标,与  $\alpha$  算法、启发式算法进行了对比分析,并在 ProM 框架下进行了实验验证。结果表明,该方法不仅可以挖掘出顺序、选择、并行、循环短循环和递归结构等流程结构,而且可以减少噪声对挖掘正确流程的影响度,计算复杂度低。当然本方法还有不足之处,对复杂流程结构或噪声含量 $\geq 20\%$ 时,本方法的日志流程挖掘能力受限,这将是今后研究的重点。

##### 参考文献:

- [1] 赵卫东,范力. 工作流挖掘研究的现状与发展[J]. 计算机集成制造系统, 2008, 11(12): 2289 - 2296.
- [2] RAKESH A, DIMITRIOS G, FRANK L. Mining process models from workflow logs [C]// Proceedings of the 6th International Conference on Extending Database Technology: Advances in Database Technology. Berlin: Springer-Verlag, 1998: 469 - 483.
- [3] van der AALST W M P, AJMM W, MARUSTER L. Workflow mining: Discovering process models from event logs [J]. IEEE Transac-

##### 参考文献:

- [1] HÄTÖNEN K, KLEMETTINEN M, MANNILA H, *et al.* TASA: Telecommunications alarm sequence analyzer or how to enjoy faults in your network [C]// IEEE/IFIP 1996 Network Operations and Management Symposium (NOMS'96). Washington, DC: IEEE, 1996: 520 - 529.
- [2] 肖海林. 网络告警关联规则挖掘系统的研究与设计[D]. 成都: 电子科技大学, 2007.
- [3] 朱扬勇, 熊赞. DNA 序列数据挖掘技术[J]. 软件学报, 2007, 18(11): 2766 - 2781.
- [4] AGRAWAL R C, AGRAWAL C C, PRASAD V V V. Depth first generation of long patterns [C]// Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM press, 2000: 108 - 118.
- [5] AGRAWAL R, IMIELINSKI T, SWAMI A. Mining association rules between sets of items in large databases [C]// Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data. Washington, DC: ACM press, 1993: 207 - 216.
- [6] HAN J W, PEI J, YIN Y W. *et al.* Mining frequent patterns without candidate generation: A frequent-pattern tree approach [J]. Data Mining and Knowledge Discovery, 2004, 8(1): 53 - 87.
- [7] CHRISTIAN B. An implementation of the FP - growth algorithm [C]// Proceedings of the 1st International Workshop on Open Source Data Mining. New York: ACM Press, 2005: 1 - 5.
- [8] LU G M, LU H J, YU J X, *et al.* AFOPT: An efficient implementation of pattern growth approach [EB/OL]. [2010 - 08 - 30]. <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.6.1757&rep=rep1&type=pdf>.
- [9] 周钦亮, 李玉忱, 公爱国. 一种新的高效生成 FpTree 条件模式基的算法[J]. 计算机应用, 2006, 26(6): 1118 - 1121.
- [10] 范明, 范宏建. 数据挖掘导论[M]. 北京: 人民邮电出版社, 2006: 237 - 240.

tions on Knowledge and Data Engineering, 2004, 16(9): 1128 - 1142.

- [4] AJMM W, van der AALST W M P. Process mining: Discovering workflow models from event-based data [C]// BNAIC2001: Proceedings of the 13th Belgium-Netherlands Conference on Artificial Intelligence. Maastricht: BNVKI, 2001: 283 - 290.
- [5] AJMM W, van der AALST W M P. Rediscovering workflow models from event-based data using little thumb [J]. Integrated Computer-Aided Engineering, 2003, 10(2): 151 - 162.
- [6] COOK J E, WOLF A L. Discovering models of software process from event-based data [J]. ACM Transactions on Software Engineering and Methodology, 1998, 7(3): 215 - 249.
- [7] van DONGEN B, de MEDEIROS A, VERBEEK H, *et al.* The prom framework: A new era in process mining tool support [C]// Proceedings of Application and Theory of Petri Nets 2005. Berlin: Springer-Verlag, 2005: 444 - 454.
- [8] ROZINAT A, van der AALST W M P. Conformance testing: Measuring the alignment between event logs and process models [EB/OL]. [2010 - 05 - 10]. [http://cms.ieis.tue.nl/Beta/Files/WorkingPapers/Beta\\_wp144.pdf](http://cms.ieis.tue.nl/Beta/Files/WorkingPapers/Beta_wp144.pdf).
- [9] ALVES A K, de MEDEIROS A, UNTHERR C W G. Process mining: Using CPN tools to create test logs for mining algorithms [EB/OL]. [2009 - 08 - 11]. <http://is.tm.tue.nl/research/processmining/tools/ProM/cpnToolConverter.zip>.