

引用格式:王媛媛,杜玉越. 基于扩展足迹矩阵的一致性检测[J]. 山东科技大学学报(自然科学版), 2018, 37(2):9-15.

WANG Yuanyuan, DU Yuyue. Conformance checking based on extended footprint matrix[J]. Journal of Shandong University of Science and Technology (Natural Science), 2018, 37(2):9-15.

基于扩展足迹矩阵的一致性检测

王媛媛,杜玉越

(山东科技大学 计算机科学与工程学院,山东 青岛 266590)

摘要:一致性检测是过程挖掘研究的重要内容,现有的一致性检测方法不足以找到准确的偏差位置。本文以 Petri 网作为描述过程模型的形式化方法,根据基于日志的次序关系重新定义了一种基于日志的扩展次序关系,并以此定义了基于日志的扩展足迹矩阵和基于模型的扩展足迹矩阵。通过对比两个扩展足迹矩阵得到日志和模型之间精准的偏差位置。最后,通过索赔申请处理的实例验证了方法的有效性。

关键词:一致性检测;过程挖掘;扩展次序关系;扩展足迹矩阵;偏差位置

中图分类号:TP311

文献标志码:A

文章编号:1672-3767(2018)02-0009-07

DOI: 10.16452/j.cnki.sdkjzk.2018.02.002

Conformance Checking Based on Extended Footprint Matrix

WANG Yuanyuan, DU Yuyue

(College of Computer Science and Engineering, Shandong University of
Science and Technology, Qingdao, Shandong 266590, China)

Abstract: The methods of existing conformance checking as an important part of process mining research are not sufficient to find the correct deviation location. Based on Petri net as the formalization ways of describing process model, this paper redefined the log-based extended ordering relations via log-based ordering relations and defined the log-based extended footprint matrix and the model-based extended footprint matrix. The precise deviation locations between the log and the model are obtained by comparing two extended footprint matrices. Finally, the validity of the method was verified by an example of the handling of compensation requests.

Key words: conformance checking; process mining; extended ordering relations; extended footprint matrix; deviation location

近年来,过程挖掘技术快速发展,很多企业已将过程挖掘功能添加到其产品中。过程挖掘主要包括三个方面:过程发现、一致性检测和过程增强。过程发现是根据事件日志生成模型,并不使用任何先验信息;一致性检测是将一个已知的过程模型与这个模型的事件日志进行对比;过程增强则是使用相关事件日志来扩展或者改进现有过程模型^[1]。

收稿日期:2017-09-14

基金项目:国家自然科学基金项目(61170078, 61472228);山东省泰山学者建设工程专项;山东省自然科学基金项目(ZR2014FM009);山东省优秀中青年科学家科研奖励基金项目(BS2015DX010);山东科技大学科技创新项目(SDKDYC170224)

作者简介:王媛媛(1993—),女,山东潍坊人,硕士研究生,主要从事过程挖掘和 Petri 网方面研究. E-mail:2419589136@qq.com

杜玉越(1960—),男,山东聊城人,教授,博士生导师,CCF 高级会员,主要从事软件工程、形式化技术和 Petri 网方面研究,本文通信作者. E-mail:yydu001@163.com

在系统和实际过程中过程模型和事件日志经常有不匹配的情况,造成这种情况的原因有多种。比如,系统不是为特定企业开发的、外部环境影响、业务过程变化迅速或者过程中不同利益相关人员之间的需求可能出现冲突等。一致性检测的目标是找到过程模型和事件日志之间的差异,确保系统和实际业务过程保持良好的合规性^[2]。通过分析实际过程和诊断差异,可以从中获得如何改进模型的支持。

文献[1]中针对一致性检测提出了托肯重演、对比足迹等方法。托肯重演^[1-2]是利用日志在模型上进行重演,记录所有丢失的和遗留的托肯来计算拟合度。对比足迹则是根据基于日志的次序关系^[1]生成足迹,也就是表示因果依赖关系的矩阵来展示事件日志和过程模型的特征,然后通过对比得到它们之间的不同,但该方法没有给出如何寻找偏差位置。基于 α 算法生成的次序关系也并不能涵盖所有事件之间的依赖关系,因此在进行一致性检测的时候,不能准确的找出所有偏差和识别偏差域。文献[3]通过生成校准^[4-6]并添加基于日志动作位置的定义,进行一致性检测与后续的模型修正。然而这种方法只能得到大概的偏差位置,还需要进一步精确。文献[7]中提出校准动作与过程模型的映射,通过校准动作映射提供的信息正确识别出错误模型中的问题,也是要利用校准进行一致性检测。文献[8]提出了一种新型日志 Token Log 的符合性检查方法,使用日志内容检查模型结构正确性与使用模型结构检查日志内容完整性标准,但是需要进行双向检查。因此,本文利用矩阵^[9]的概念提出一种基于扩展足迹矩阵的一致性检测方法,不需要利用校准,而是扩展活动间的依赖关系以得到日志和模型的扩展足迹矩阵,进而更精确地识别偏差位置。

1 基本概念

Petri 网是一种用于系统分析与建模的工具,被广泛应用于多个领域,特别便于描述系统的顺序、并发、冲突以及同步等关系^[10],既能描述系统的结构,又能模拟系统的运行。本节简要介绍 Petri 网的基本定义及基于日志的次序关系^[1]相关概念。

定义 1(迹,事件日志^[11-12]) 设 \mathcal{A} 是所有活动的集合, $A \subseteq \mathcal{A}$ 。若活动序列 $\sigma \in A^*$, A^* 表示集合 A 上有限序列的集合,则称 σ 是一条迹。若 $\exists L \in B(A^*)$ 是迹的一个多重集,称 L 为一个事件日志。用 $\&(\sigma)$ 表示迹 σ 中所有活动构成的集合。

定义 2(前集,后集^[11-13]) 设 $N = (P, T; F)$ 为一个网。对于 $\forall x \in P \cup T$, 令

$$\cdot x = \{y | y \in P \cup T \wedge (y, x) \in F\},$$

$$x \cdot = \{y | y \in P \cup T \wedge (x, y) \in F\},$$

称 $\cdot x$ 为 x 的前集或输入集, $x \cdot$ 为 x 的后集或输出集。

定义 3(Petri 网^[11-13]) 一个四元组 $PN = (P, T; F, M)$ 称作 Petri 网,当且仅当

1) $N = (P, T; F)$ 为一个网;

2) 映射 $M: P \rightarrow \{0, 1, 2, \dots\}$ 为网 N 的一个标识(marking)。

定义 4(变迁发生规则) 一个四元组 $PN = (P, T; F, M)$ 是一个标识 Petri 网,具有下列变迁发生规则:

1) 变迁 $t \in T$ 在标识 M 有发生权或称为使能的(enabled),当且仅当对 $\forall p \in P: p \in \cdot t : M(s) \geq 1$,记作 $M[t >]$ 。

2) 若 $M[t >]$,则在标识 M 下发生变迁 t 后得到一个新的标识 M' ,记作 $M[t > M']$,且对 $\forall p \in P$,

$$M'(p) = \begin{cases} M(p)-1, & \text{若 } p \in \cdot t - t \cdot, \\ M(p)+1, & \text{若 } p \in t \cdot - \cdot t, \\ M(p), & \text{其他。} \end{cases}$$

从 M 可达的一切标识的集合记为 $R(M)$,约定 $M \in R(M)$ 。

定义 5(基于日志的次序关系^[1]) 使用 L 表示一个基于 \mathcal{A} 的事件日志,即 $L \in B(\mathcal{A}^*)$ 。令 $a, b \in \mathcal{A}$,那么:

1) $a >_L b$ 当且仅当存在一个轨迹 $\sigma = \langle t_1, t_2, t_3, \dots, t_n \rangle$, $i \in \{1, \dots, n-1\}$ 使得 $\sigma \in L$, $t_i = a$ 并且 $t_{i+1} = b$;

2) $a \rightarrow_L b$ 当且仅当 $a >_L b$ 并且 $b \not>_L a$;

- 3) $a \#_L b$ 当且仅当 $a \not\succ_L b$ 并且 $b \not\succ_L a$;
 4) $a \parallel_L b$ 当且仅当 $a >_L b$ 并且 $b >_L a$ 。

2 扩展足迹矩阵

过程模型可以通过事件日志挖掘出来,并越来越倾向于对真实生活中自然语言的处理和挖掘^[14-15]。给定事件日志和过程模型,则可以根据两者之间的扩展足迹来对比它们之间的差异。文献[1]基于 α 算法提出的基于日志的次序关系,虽然可以将基本的活动间的关系表示出来,但是对于不是直接跟随关系或者不是直接因果关系的活动无法表示两者之间深层次的关系,这就导致事件日志和过程模型在进行一致性检测的时候很容易将活动之间的关系漏掉,以至于检测出的事件日志和过程模型之间的偏差位置并不精确。因此,本节针对此问题进行了改进,提出了基于日志的扩展次序关系以及扩展足迹矩阵的相关定义,扩展后可以将事件日志中活动之间的关系更完备的表达出来。在定义6中,扩展了间接因果关系和并行关系。下面首先给出基于日志的扩展次序关系。

定义6(基于日志的扩展次序关系) 设 L 表示一个基于 A 的事件日志,即 $L \in B(A^*)$ 。 $\sigma \in L$ 是日志中的一条迹, $a, b \in \&(\sigma)$,那么:

- 1) 直接跟随关系 $>: a > b$ 当且仅当存在一个轨迹, $\sigma = \langle t_1, t_2, t_3, \dots, t_n \rangle, i \in \{1, \dots, n-1\}$ 使得 $\sigma \in L$, $t_i = a$ 并且 $t_{i+1} = b$;
- 2) 直接因果关系 $\rightarrow: a \rightarrow b$ 当且仅当任意 $a, b \in \&(\sigma), a > b$ 并且 $b \not\succ a$;
- 3) 间接因果关系 $\rightarrow_i: a \rightarrow_i b$ 当且仅当存在 $a, t_1, \dots, t_n, b \in \&(\sigma), a \not\succ b$ 且 $b \not\succ a$, 有 $a \rightarrow t_1 \rightarrow \dots \rightarrow t_n \rightarrow b$;
- 4) 并行关系 $\parallel: a \parallel b$ 当且仅当存在 $\sigma_1, \sigma_2 \in L, a, b \in A$, 对于 $a, b \in \&(\sigma_1): a > b$ 或者 $a > \dots > b$, 并且对于 $a, b \in \&(\sigma_2): b > a$ 或者 $b > \dots > a$ 。
- 5) 排他关系: $a \# b$ 当且仅当任意 $a, b \in \&(\sigma), a \not\succ b$ 并且 $b \not\succ a$ 。

$>_L$ 关系包含了日志 L 中所有紧邻的活动对,是关于日志 L 中所有迹的完备的直接跟随关系。例如日志 $L = [\langle a, b, c, d \rangle, \langle a, c, b, d \rangle, \langle a, e, d \rangle]$, 在轨迹 $\langle a, b, c, d \rangle$ 中 b 紧跟在 a 之后,所以有 $a > b$ 。 \rightarrow_L 关系包含了日志 L 中全部具有直接因果关系的活动对。例如,在日志 L 中 b 直接跟随于 a ,但 a 不直接跟随于 b ,所以有 $a \rightarrow b$ 。 \rightarrow_{iL} 关系包含了日志 L 中全部具有间接因果关系的活动对。例如,在日志 L 中 $a \cdots d$,但 a 不直接跟随于 d ,所以有 $a \rightarrow_i d$ 。由于原次序关系并没有深层次的挖掘活动间的关系,造成在进行一致性检测的时候很多偏差识别不出来。所以本文在添加间接因果关系之后,可以根据日志和模型间直接和间接因果关系的变化推导出偏差位置,比如一个新活动的增加势必会使原本模型中的直接因果关系变成间接因果关系。 \parallel_L 表示日志 L 中所有具有并发关系的活动对。例如,在日志 L 中,迹 $\langle a, b, c, d \rangle$ 和 $\langle a, c, b, d \rangle$ 中有 $b > c$ 且 $c > b$,即 b 和 c 互相存在直接跟随关系的情况,所以有 $b \parallel c$ 。在并行关系中并非只有 $a > b$ 并且 $b > a$ 的情况,还可能存在两者之间不是直接跟随关系,而是并不存在因果依赖关系但是必定都在同一条迹中发生的情况,所以这种定义方式并不严谨。因此,本文根据存在并行关系的两个活动的所有可能情况作了扩展。 $\#_L$ 表示日志 L 中所有具有排他关系的活动对的集合。例如,在日志 L 的所有迹中, b, e 都不能同时发生,所以有 $b \# e$ 。

对于 $\forall a, b \in A$, 有且只有一种确定的关系。在日志 L 中,一定有 $a \rightarrow b, a \rightarrow_i b, a \parallel b$ 或者 $a \# b$ 。也就是说,任一活动对一定满足这4种关系中的一种,且任意两个活动之间的依赖关系都是确定的。因此可以用一个扩展足迹矩阵来描述日志的足迹^[1],即表示活动间依赖关系的矩阵。下面给出基于日志的扩展足迹矩阵和基于模型的扩展足迹矩阵的相关定义。

定义7(基于日志的扩展足迹矩阵) 设 L 是一个基于 A 的事件日志,即 $L \in B(A^*)$ 。 $LEFM=L[n][n]$ 是一个基于日志的扩展足迹矩阵, $n=|A|$ 。 $|A|$ 表示集合 A 中所有元素的个数。对于 $l_i, l_j \in A, i \in \{1, \dots, n-1\}, j \in \{i+1, \dots, n\}$, $L[i][j]$ 表示活动 l_i, l_j 之间基于日志的扩展次序关系。

定义8(基于模型的扩展足迹矩阵) 设 $PN=(P, T; F, M)$ 是一个 Petri 网。 $MEFM=M[n][n]$ 是一个基于模型的扩展足迹矩阵。对于 $m_i, m_j \in T, i \in \{1, \dots, n-1\}, j \in \{i+1, \dots, n\}$, $M[i][j]$ 表示 m_i, m_j 之间

基于模型 PN 重演记录的执行序列的扩展次序关系。

例 1 图 1 展示了一个用 Petri 网表示的过程模型 N_1 , 这个模型描述了一家航空公司处理索赔申请的过程。该过程从处理一个申请开始, 为了方便, 用标签为 a (register request)的变迁表示。当申请很复杂或可疑时, 执行变迁 b (examine thoroughly), 当申请比较简单时, 只需要执行非正式的检查 c (examine casually)即可。这两个变迁是选择关系, 即一方执行都不会再执行另一方。变迁 d (check ticket)代表一项查看顾客是否由自个提出申请的行政检查, 变迁 e (decide)代表着可能做出的决策。最后支付所申请的赔偿 g (pay compensation)或者拒绝赔偿 h (reject request), 或者需要更多的检查将会发生变迁 f (reinitiate request)。对于变迁 f 发生的情况, 整个过程将会重新回到选择结构(b, c)的位置继续执行。实际中整个过程可能会发生多次迭代, 会在支付或者拒绝赔偿后结束。

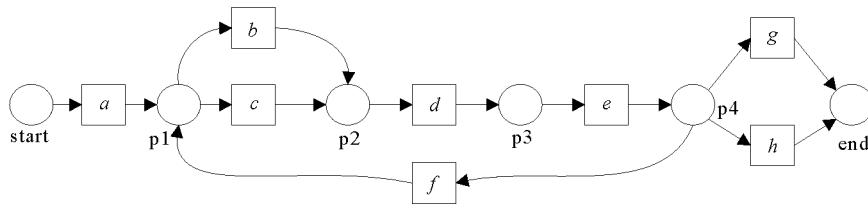


图 1 一个描述索赔申请处理过程的 Petri 网 N_1

Fig. 1 A Petri net N_1 modeling the handling of compensation requests

例 2 过程模型 N_1 如图 1 所示, 日志 $L = \{<a, c, d, e, h>, <a, b, d, e, h>, <a, d, c, e, g>, <a, d, b, e, h>, <a, c, d, e, f, b, d, e, h>, <a, b, d, e, f, d, b, e, g>\}$, 日志 L 从实际流程中得到。下面列出基于日志的扩展次序关系, 并得出基于日志的扩展足迹矩阵和基于模型的扩展足迹矩阵。

日志 L 的 6 条迹中包含的所有扩展次序关系如下所示:

1) 直接跟随关系 $>$:

$$\begin{aligned} >_{\sigma 1} &= \{(a, c), (c, d), (d, e), (e, h)\}; \\ >_{\sigma 2} &= \{(a, b), (b, d), (d, e), (e, h)\}; \\ >_{\sigma 3} &= \{(a, d), (d, c), (c, e), (e, g)\}; \\ >_{\sigma 4} &= \{(a, d), (d, b), (b, e), (e, h)\}; \\ >_{\sigma 5} &= \{(a, c), (c, d), (d, e), (e, f), (f, b), (b, d), (d, e), (e, h)\}; \\ >_{\sigma 6} &= \{(a, b), (b, d), (d, e), (e, f), (f, d), (d, b), (b, e), (e, g)\}; \\ >_L &= \{(a, b), (a, c), (a, d), (b, d), (b, e), (c, d), (c, e), (e, h), (e, g), (e, f), (f, b), (f, d)\} \end{aligned}$$

2) 直接因果关系 \rightarrow :

$$\rightarrow_L = \{(a, b), (a, c), (a, d), (b, e), (c, e), (d, e), (e, g), (e, f), (e, h), (f, b), (f, d)\};$$

3) 间接因果关系 \rightarrow_i :

$$\rightarrow_{iL} = \{(a, e), (a, g), (a, h), (a, f), (b, g), (b, h), (c, f), (c, g), (c, h), (d, g), (d, h), (f, g), (f, g)\};$$

4) 排他关系 #:

$$\#_L = \{(b, c), (g, h)\};$$

5) 并行关系 || :

$$\parallel_L = \{(b, d), (c, d)\}$$

由此得到基于日志 L 的扩展足迹矩阵如下:

a	b	c	d	e	f	g	h	
$\rightarrow \rightarrow \rightarrow \rightarrow i \rightarrow i \rightarrow i \rightarrow i$								a
#	\parallel	\rightarrow	\leftarrow	$\rightarrow i$	$\rightarrow i$			b
	\parallel	\rightarrow	$\rightarrow i$	$\rightarrow i$	$\rightarrow i$			c
		\rightarrow	\leftarrow	$\rightarrow i$	$\rightarrow i$			d
				\rightarrow	\rightarrow			e
					$\rightarrow i$	$\rightarrow i$		f
						#		g
								h

重演模型 N_1 记录执行序列,产生的基于模型的扩展足迹矩阵如下:

a	b	c	d	e	f	g	h	
$\rightarrow \rightarrow \rightarrow \rightarrow i \rightarrow i \rightarrow i \rightarrow i$								a
#	\rightarrow	$\rightarrow i$	\leftarrow	$\rightarrow i$	$\rightarrow i$			b
	\rightarrow	$\rightarrow i$	$\rightarrow i$	$\rightarrow i$	$\rightarrow i$			c
	\rightarrow	$\rightarrow i$	$\rightarrow i$	$\rightarrow i$	$\rightarrow i$			d
			\rightarrow	\rightarrow	\rightarrow			e
				$\rightarrow i$	$\rightarrow i$			f
						#		g
								h

通过得到的日志 L 的扩展足迹矩阵和模型 N_1 的扩展足迹矩阵可以看出,事件日志中有些活动的依赖关系相比过程模型有了改变,下一节将通过对两者之间的变化识别出事件日志和过程模型之间的偏差位置。

3 一致性检测

第2节中已经得到了基于日志的扩展足迹矩阵和基于模型的扩展足迹矩阵,本节通过对两种扩展足迹矩阵,并在此基础上定义对比扩展足迹矩阵的概念,通过求得的对比扩展足迹矩阵,可以识别出事件日志和过程模型之间精确的偏差位置。

3.1 对比扩展足迹矩阵

定义9(对比扩展足迹矩阵) 设 L 是一个基于 A 的事件日志,即 $L \in B(A^*)$, $PN = (P, T; F, M)$ 是一个 Petri 网。 $LEFM = L[n][n]$ 是一个基于日志的扩展足迹矩阵, $n = |A|$ 。 $MEFM = M[n][n]$ 是一个基于模型的扩展足迹矩阵。 $CEFM = C[n][n]$ 是一个对比扩展足迹矩阵。对于 $l_i, l_j \in A, i \in \{1, \dots, n-1\}, j \in \{i+1, \dots, n\}$, $C[i][j]$ 表示扩展足迹矩阵 $LEFM$ 与 $MEFM$ 中不一致的元素, $C[i][j] = L[i][j] - M[i][j]$ 。即活动 l_i, l_j 之间在日志中并且不在模型中的依赖关系。

定义9中,对比扩展足迹矩阵中保留了两个扩展足迹矩阵中不一致的元素,即模型中存在偏差的部分。下面给出对比扩展足迹矩阵的算法。

算法1(对比扩展足迹矩阵算法)

设 $LEFM = L[n][n]$ 是一个基于日志的扩展足迹矩阵, $MEFM = M[n][n]$ 是一个基于模型的扩展足迹矩阵, $CEFM = C[n][n]$ 是一个对比扩展足迹矩阵。

输入: $LEFM, MEFM$

输出:对比扩展足迹矩阵 $CEFM$

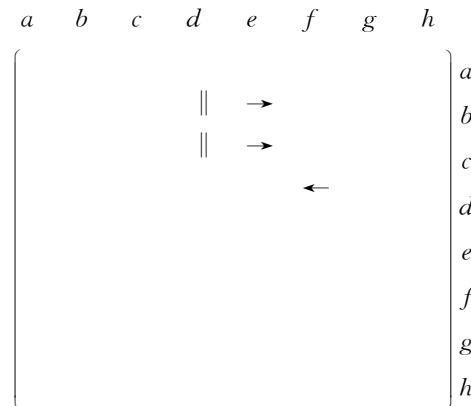
① $L[i][j]$ 表示日志的扩展足迹矩阵 LEFM 中的元素, $M[i][j]$ 表示模型的扩展足迹矩阵 MEFM 中的元素, $C[i][j]$ 用来存储不一致元素, 初始设为空。

② 根据定义 9, 逐步遍历 $L[n][n]$ 和 $M[n][n]$, 若 $L[i][j] = M[i][j]$, 跳过;

③ 若 $L[i][j] \neq M[i][j]$, 更新矩阵 CEFM , $C[n][n] \leftarrow C[n][n] \cup \{L[i][j]\}$; // 将 $L[i][j]$ 保存到矩阵 CEFM 中;

④ RETURN CEFM

例 3 过程模型 N_1 如图 1 所示, 日志 $L = \{\langle a, c, d, e, h \rangle, \langle a, b, d, e, h \rangle, \langle a, d, c, e, g \rangle, \langle a, d, b, e, h \rangle, \langle a, c, d, e, f, b, d, e, h \rangle, \langle a, b, d, e, f, d, b, e, g \rangle\}$, 日志 L 从实际流程中得到。由例 2, 得出对比扩展足迹矩阵。



3.2 定位偏差位置

例 4 过程模型 N_1 如图 1 所示, 日志 $L = \{\langle a, c, d, e, h \rangle, \langle a, b, d, e, h \rangle, \langle a, d, c, e, g \rangle, \langle a, d, b, e, h \rangle, \langle a, c, d, e, f, b, d, e, h \rangle, \langle a, b, d, e, f, d, b, e, g \rangle\}$, 日志 L 从实际流程中得到。通过例 2、例 3 求得对比扩展足迹矩阵, 可以很清晰地看到: 日志中变迁 b (examine thoroughly) 和 c (examine casually) 构成的选择结构跟变迁 d (check ticket) 应该是并行关系, 但原模型中却显示的是因果关系。日志中 b (examine thoroughly), c (examine casually), d (check ticket) 与 e (decide) 是直接跟随关系, 而在原模型中错误的将 b (examine thoroughly), c (examine casually) 与 e (decide) 表示成间接跟随关系。因为 b, c, d 三个变迁结构的改变, 在日志中存在循环时, f (reinitiate request) 与 d (check ticket) 的关系在模型中是间接因果关系, 日志中是直接因果关系, 说明实际返回变迁 f 与并行结构 $\{(b, c), d\}$ 是直接因果关系, 之间存在两条流关系直接相连。除此之外, 日志和模型之间直接因果关系和间接因果关系的变化对于添加新活动的情况也能很好的识别出来。利用这些信息可以很准确的对模型进行修正。

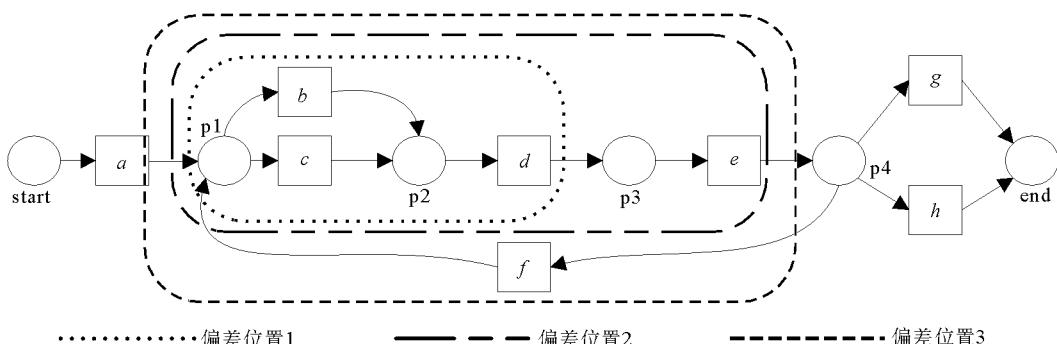


图 2 图 1 中的 N_1 的偏差位置

Fig. 2 The deviation location of N_1 in figure 1

4 结束语

一致性检测是过程挖掘研究的重要内容,本文提出了一种基于扩展足迹矩阵的一致性检测方法。具体工作是对已有的基于日志的次序关系进行扩展,得到基于日志的扩展次序关系,进而得到两种扩展足迹矩阵。最后对比两个扩展足迹矩阵之间的差异得到精确的偏差位置。本文所提出的方法不需要利用校准,并且能够更精确地识别偏差位置,尤其能够更好地识别模型存在结构错误的情况。下一步的工作是基于以上工作对模型中的偏差位置进行修正,得到正确的过程模型。

参考文献:

- [1] VAN DER AALST W M P. Process mining:Discovery,conformance and enhancement of business process[M]. Berlin, Heidelberg:Springer,2011:64-162.
- [2] VAN DER AALST W M P,ADRIANSYAH A,VAN DONGEN B. Replaying history on process models for conformance checking and performance analysis[J]. WIRES Data Mining and Knowledge Discovery,2012,2(2):182-192.
- [3] FAHLAND D,VAN DER AALST W M P. Model Repair: Aligningprocess models to reality[J]. Information Systems,2015, 47(1):220-243.
- [4] WANG L,DU Y X,LIU W. Aligning observed and modeled behavior based on workflow decomposition[J]. Enterprise Information Systems,2017,11(8):1207-1227.
- [5] ADRIANSYAH A. Aligning observed and modeled behavior[D]. Eindhoven: Technische Universiteit Eindhoven,2014:122-149.
- [6] ADRIANSYAH A, MUÑOZ-GAMA J, CARMONA J, et al. Alignment based precision checking[C]//Business Process Management Workshops. Berlin, Heidelberg:Springer,2013:137-149.
- [7] 王路,杜玉越.一种基于校准的模型问题域识别方法[J].山东科技大学学报(自然科学版),2015,34(1):42-46.
WANG Lu,DU Yuyue. An alignment-based identifying method of the problem areas within process models[J]. Journal of Shandong University of Science and Technology (Natural Science),2015,34(1):42-46.
- [8] 李传艺,葛季栋,胡海洋,等.一种基于 Token Log 的符合性检查方法[J].软件学报,2015,26(3):509-532.
LI Chuanyi,GE Jidong,HU Haiyang,et al. Method for conformance checking based on Token Log[J]. Journal of Software, 2015,26(3):509-532.
- [9] SUN Y N,DU Y Y,LI M Z. A repair of workflow models based on mirroring matrices[J]. International Journal of Parallel Programming,2016,45(4):1-20.
- [10] 蒋昌俊. Petri 网的行为理论及应用[M].北京:高等教育出版社,2003:19-26.
- [11] SUN S,AKHIL K,JOHN Y. Merging workflows: A new perspective on connecting business processes[J]. Decision Support Systems,2006,42(2):844-858.
- [12] 查海平,王建民,闻立杰.一种 Petri 网模型完备日志生成算法[J].系统仿真学报,2007,19(A01):271-274.
ZHA Haiping,WANG Jianmin,WEN Lijie. An algorithm of complete log generation for Petri net models[J]. Journal of System Simulation,2007,19(A01):271-274.
- [13] MURATA T. Petri nets:Properties,analysis and applications[J]. Proceedings of the IEEE,1989,77(4):541-580.
- [14] AALST W V D. Process mining:Overview and opportunities[J]. ACM Transactions on Management Information Systems, 2012,3(2):1-17.
- [15] AALST W V D,DONGEN B F V. Discovering Petri nets from event logs[J]. Transactions on Petri Nets and Other Models of Concurrency,2013,7:372-422.