

# 基于行为校验与偏差组合的过程模型修复

张力雯<sup>1</sup>, 方贤文<sup>1+</sup>, 邵叱风<sup>1</sup>, 王丽丽<sup>1,2</sup>

(1. 安徽理工大学 数学与大数据学院, 安徽 淮南 232001;

2. 同济大学 嵌入式系统与服务计算教育部重点实验室, 上海 201804)

**摘要:**模型修复通过对偏差行为的合理调整,改善事件日志与业务流程之间的一致性性能。采用自循环插入方式对日志中可观测的偏差活动进行模型修复,将优先考虑适合度的提升而忽视精度。为获得事件日志与过程模型之间不可回放的行为模式,根据行为关系将其可达活动图划分为若干个片段并进行服从性校验。利用回放过程中行为模式所产生的最优对齐检测偏差的发生、位置以及潜在行为关系,将具有直接跟随关系的偏差元素构建为可修复的子结构,从而通过减少事件日志中的偏差个数而改善精度。通过实验使用 M-repair 插件在不同数据集上进行评估,结果表明该方法相较于现存方法在保证适合度的前提下可显著提升精度。

**关键词:**模型修复;可达活动图表;最优对齐;直接跟随活动;偏差子结构

**中图分类号:**TP391.9 **文献标识码:**A

## Process model repair based on behavior checking and combination of deviations

ZHANG Liwen<sup>1</sup>, FANG Xianwen<sup>1+</sup>, SHAO Chifeng<sup>1</sup>, WANG Lili<sup>1,2</sup>

(1. School of Mathematics and Big Data, Anhui University of Science and Technology, Huainan 232001, China;

2. Key Laboratory of Embedded System and Service Computing, Ministry of Education, Tongji University, Shanghai 201804, China)

**Abstract:** The consistency performance between the event log and the business process can be improved by model repair reasonably adjusting the deviation behaviors. The deviations observed the event log are repaired in a self-loop insert way, so the improvement of fitness will be prioritized over precision. To obtain the non-replayed behavior mode between event log and process model, the reachable activity structure was divided into several fragments and the conformance was checked according to the behavior relationship. The optimal alignment produced by the behavior patterns during replay was used to detect the occurrences, locations and potential behavior relationships of deviations. Furthermore, the directly following deviation elements were constructed into repairable substructure. Therefore, the precision could be improved by reducing the number of deviations in the event log. M-repair plug-in was applied to evaluate different data sets, and the result showed that the precision was significantly promoted ensuring the fitness compared with the existing method.

**Keywords:** model repair; reachable activity graph; optimal alignment; directly following deviations; deviation sub-structure

收稿日期:2020-12-28。Received 28 Dec. 2020.

基金项目:国家自然科学基金资助项目(61572035,61402011);安徽省高校领军骨干人才资助项目(2020-1-12);安徽省自然科学基金资助项目(2008085QD178);安徽省学术和技术带头人资助项目(2019H239);安徽省高校优秀人才支持计划资助项目(gxyqZD2020020);嵌入式系统与服务计算教育部重点实验开放课题(ESSCKF2018-04)。**Foundation items:** Project supported by the National Natural Science Foundation, China (No. 61572035, 61402011), the Leading Backbone Talent Foundation for Anhui Provincial Higher Education, China (No. 2020-1-12), the Natural Science Foundation of Anhui Province, China (No. 2008085QD178), the Anhui Provincial Academic and Technical Leader Foundation, China(No. 2019H239), the Anhui Provincial College Excellent Young Talents Fund, China (No. gxyqZD2020020), and the Open Foundation of the Key Laboratory of Embedded System and Service Computing Ministry of Education, China (No. ESSCKF2018-04).

## 0 引言

### (1) 研究背景

过程挖掘技术是从信息系统所记录的事件日志中提取知识发现过程模型,并通过提供技术和工具对其实现性能上的改善<sup>[1]</sup>。过程挖掘技术主要从过程发现、服从性校验和过程完善<sup>[2]</sup>3个方面对业务流程进行分析,其中过程发现及服从性校验最为重要<sup>[3]</sup>。模型修复是介于过程发现与服从性校验之间的一种过程挖掘的新类型,主要考虑使得事件日志无法回放于过程模型的部分行为,并使修复后的过程模型尽可能与初始模型相似<sup>[4]</sup>。模型修复适用于一致性校验、演变过程监测、定制化需求3种情况,本文主要根据一致性校验检测到的无法回放行为对业务流程进行修复<sup>[5]</sup>。一致性校验可以被应用于不同的设置中,主要包括合规性审计、模型维护以及自动化的过程发现。合规性审计只可以检测出事件日志无法回放于过程模型的行为,即不适合行为<sup>[6]</sup>;模型维护是检测所有偏差行为的设置,既可以诊断事件日志无法回放于模型的行为也可以发现模型中所特有的行为,即不适合行为与额外行为<sup>[7]</sup>;自动化的过程发现是根据一致性校验的输出结果对过程模型进行调整,通过移除日志中没有记录但模型中可以捕获的行为或者添加仅日志中可观测的行为来实现对初始模型的自动化调整。本文通过并发关系、序列关系、冲突关系划分事件日志与过程模型的可达活动图表,并基于行为关系的一致性校验识别它们之间可回放与不可回放的行为模式。

模型修复主要针对事件日志无法回放于过程模型上的不适合行为进行处理,当前技术采用自循环插入/跳过的方式优先考虑适合度的改善<sup>[8-9]</sup>。对齐事件日志与过程模型,并检测到它们之间尽可能少的偏差元素,更有利于对适合度的分析<sup>[10]</sup>。通过最优对齐检测偏差成本,并将多个经典问题的解决方案转换为搜索修复建议的方法,即基于偏差成本与修复资源间的单调性在适合度与计算量之间进行权衡<sup>[11]</sup>。根据一致性校验的四个性能参数从不同角度介绍过程模型的修复方法,并将满足并发关系的两个直接跟随偏差组合为子结构<sup>[12]</sup>。这种方法能够在保证适合度的前提下,对精度起到一定的改善。但其没有定义对应的偏差检测方法,因此无法适用于各种不同的情况。由于当前技术往往需要牺牲一定的精度来换取适合度的提升,容易出现欠拟合的

情况<sup>[13-14]</sup>。产生上述问题的原因主要是修复工作只基于偏差的成本和位置信息来进行,而不考虑它们之间的行为关系。本文是对之前研究工作的延伸,其将所有满足特定行为关系的直接跟随偏差构建为子结构。由于拟合度直接反映事件日志是否完整地回放于过程模型,需要在保证拟合度的前提下尽可能提高精度的值。

### (2) 相关工作

本文所提方法与以下两个方面的技术有着紧密联系:①业务流程的服从性校验;②插入自循环结构的简单模型修复<sup>[15]</sup>。

根据偏差类型的不同将当前服从性校验方法分为两类:①基于事件日志与过程模型之间的最优对齐检测偏差的发生次数与位置;②基于事件日志与过程模型之间行为关系发现不匹配行为模式。第①类校验方法利用启发式算法与迹回放来获取有效的最优对齐<sup>[16]</sup>,并将其中所有同步移动与非同步移动的成本分别设置为“0”、“1”(含有不可视变迁 $r$ 的非同步移动成本为“0”);第②类校验方法则是通过构建事件日志与过程模型的事件结构产生它们之间的部分同步化产品,再将所有事件的执行配置集展开发现偏差元素及其潜在的行为关系。通过介绍9种不匹配模式下可能产生的偏差,分别从不同角度对不匹配模式中的隐操作进行分析<sup>[17]</sup>。本文将这两种一致性校验方法相结合,首先基于行为关系识别事件日志与过程模型相对应的行为模式(包括可回放的行为模式与不可回放的行为模式),并利用由此产生的最优对齐检测出修复操作所需要的偏差完整信息<sup>[18]</sup>。这种完整信息不仅包含偏差的发生和位置,还包括不可回放模式中直接跟随偏差之间的行为关系。

模型修复技术旨在使修复后的过程模型能够回放事件日志,并尽可能保存初始模型中的行为。使用最小成本完成最大可能的修复以及使用手动修复尽可能减少模型上冗余的行为,仅考虑拟合度的改善而忽视精度<sup>[19-20]</sup>。定义后的扩展对齐被添加到模型上的可达标识,且过程模型上的偏差能够被核算和修复<sup>[21]</sup>。模型的修复位置通过逻辑 Petri 网和选择结构之间的转换关系确定<sup>[22]</sup>。尽管上述两种方法能够综合考虑模型修复的各种度量标准,但只适用于有选择结构的过程模型。根据6种不同算法在特定的约束条件下获得修复建议,旨在权衡适合度与修复效率之间的最优性<sup>[11]</sup>。由于这种修复方

法仅根据修复建议对每个偏差元素进行插入/跳过操作,因此会以牺牲精度为代价换取适合度的改善。基于并发行为关系构建子结构可以使精度略有提升,但仍然存在一定的局限性<sup>[12]</sup>。本文将偏差的发生、位置以及直接跟随偏差活动之间的行为关系作为模型修复的输入信息,并将若干个具有特定行为关系的直接跟随偏差构建为一个子结构。因此,可通过减少事件日志中无法被准确修复的偏差数量来提升精度值。

## 1 基本定义

**定义 1** 事件日志。  $L=(E, C, \mu, l, \mathbb{N}, \infty)$  为事件日志,其中  $E$  表示日志中所有事件,  $e$  表示单个事件元素  $\forall e \in L$ ;  $C$  表示日志中的案例编号集合,即一条迹在系统运行中发生多重次的统一标识符,  $c$  为单个案例号,  $\forall c \in C$ ;  $\mu$  为满射函数,将每个事件关联到一个案例,  $\mu: e \rightarrow c$ 。例如:  $\forall c \in C, E_1$  上所有事件的案例号都为 1,  $l$  为指定函数,将日志中的所有事件指定为相应的活动标签,  $l: E \rightarrow \mathbb{N}$ 。  $\mathbb{N}$  为日志中的活动标签集,  $l(e) \in \mathbb{N}, \forall (e) \in L$ ;  $\infty$  表示日志中所有事件之间的弱序关系,  $\infty \subseteq E \times E$ 。

**定义 2** 标签 Petri 网系统。设定一个元组  $N^*=(P, T, F, \mathbb{Z}, \lambda, M_i)$  为标签工作流网系统,其中  $P$  表示库所集,  $p$  表示网系统中的库所,  $\forall p \in P$ ;  $T$  是变迁集,  $t$  是网系统中的变迁,  $\forall t \in T$ ;  $F$  是连接所有直接跟随库所与变迁之间的流关系,  $F \subseteq (P \times T)(T \times P)$ ;  $\mathbb{Z}$  为标签 WF-net 系统上所有活动的标签集,  $\mathbb{Z} = \mathbb{Z} \cup \{\tau\}$ ;  $\lambda$  是指定函数,将所有变迁指定为相应活动的标签或  $\tau$ :  $T \leftarrow \mathbb{Z} \cup \{\tau\}, \forall \lambda(t) \in \mathbb{Z} \cup \{\tau\}$ ;  $M_i$  是网系统  $N^*$  中的初始标识,  $M_i(p) \geq 1$ 。

**定义 3** 最优对齐。对齐是事件日志与网系统  $N^*$  中执行序列与发生序列的比较集,记作  $\zeta^*$ 。令  $M_{LN^*}$  为事件日志和网系统  $N^*$  之间对齐中的同步移动集合,而  $\widehat{M}_L, \widehat{M}_{N^*}$  则分别为事件日志上的移动与过程模型上的移动,  $M_{LN^*} \cup \widehat{M}_L = L, M_{LN^*} \cup \widehat{M}_{N^*} = N^*$ 。由于对齐所检测的偏差既有可能产生于  $\widehat{M}_L$  也有可能来自于  $\widehat{M}_{N^*}$ ,将其分为以下两种类型:①插入偏差,事件日志中可以观测但过程模型中无法捕获的行为,记作  $Esert(l(e)), \forall e \in L$ ;②跳过偏差,过程模型中所特有的行为,且该行为使得事件日志无法被正确回放,记作  $Skip(\lambda(t)), \forall t \in T$ 。将一个异步移动的单位成本值设定为“1”,反之则为

“0”(含有隐变迁  $\tau$  的异步移动成本值为“0”),由此可知:  $cost(\widehat{M}_L) = cost(\widehat{M}_{N^*}) = 1, cost(M_{LN^*}) = 0$ 。最优对齐是指事件日志与过程模型之间拥有最低成本的对齐,即  $cost(\xi_{op}) \leq cost(\xi)$  且  $\forall \xi \in \xi^*, \exists \xi_{op} \in \xi^*$ 。表 1 描述了两两种异步移动产生的偏差,以及不同单位移动的成本。

表 1 单位移动的成本

成本	0	1	1
单位移动	$M_{LN^*}$	$\widehat{M}_L$	$\widehat{M}_{N^*}$

**定义 4** 不可回放的行行为模式。模型修复针对事件日志无法回放于过程模型的非一致行为进行处理,根据两个相对应模式之间行为关系的差异可将它们合并为一个不可回放的行行为模式,记作:  $B_{L/N^*} = (B_{(L/N^*) (+, <, \times)}, B_{L/N^* \subseteq}, B_{l(e)}, B_{\lambda(t)}, B_{\perp})$ 。其中:  $B_{(L/N^*) (+, <, \times)}$  是由并发、冲突、因果中任意两个行为关系所产生的不可回放行行为模式;  $B_{L/N^* \subseteq}$  表示由行为包含所产生的不可回放行行为模式,根据行为包含的严格性等级不同,可以将行为包含模式分为以下两种类型:①行行为模式的整体包含;②行行为模式的部分包含。前者是将一个完整的不可回放模式看作子结构进行修复,而后者仅考虑片段中的部分包含行为。根据行为关系的不同可将行为包含模式分为以下 3 种类型:①并发关系的行行为包含,包括整体包含与部分包含(并发/因果模式属于并发结构中的部分行为包含,反之则无需进行修复),记作  $B_{L/N^* \subseteq +}, B_{L \subseteq +}(B_{+ / <})$ ;②因果关系的行行为包含,这种情况只涉及整体包含,记作  $B_{L/N^* \subseteq <}$ ;③冲突关系的行行为包含,包括整体包含与部分包含,分别记作  $B_{L/N^* \subseteq \times}, B_{L \subseteq \times}$ 。这里的部分包含主要指事件日志包含过程模型的行为,反之则无需进行修复。例如:过程模型的行行为模式中只有一个活动  $\lambda(t)$ ,而事件日志与之相对应的行行为模式中有两个满足冲突关系的活动  $B_{L \times}(l(e), l(e'))$ ,其中  $l(e) = \lambda(t)$ 。这两个模式可以合并为一个冲突关系的部分行为包含模式:  $B_{L \subseteq \times}(B_{N^*}(\lambda(t), B_{L \times}(l(e), l(e'))))$ 。  $B_{l(e)}$  表示事件日志中可观测但过程模型中无法捕获的单个活动;  $B_{\lambda(t)}$  表示过程模型中使得事件日志无法回放的单个活动。判定事件日志与过程模型之间的两个活动匹配需同时满足标签与发生位置一致。

## 2 检测偏差的完整信息

事件日志与过程模型在实际运行中可能存在一

些不被期望的差异,而其中对日志无法回放行为的处理则是模型修复需要解决的问题。本章通过划分事件日志与过程模型的可达活动图表获得它们之间相对应的行为模式,并由此产生最优对齐来检测偏差的完整信息<sup>[23]</sup>。

## 2.1 划分可达活动图表

可达活动图表的划分是在精炼过程树的基础上的改进,采用从整体到部分层层递进的搜索方法寻找事件日志与过程模型之间相对应的片段<sup>[24-25]</sup>。根据给定约束条件在回放过程中所划分的可达活动图表可以获得事件日志与过程模型之间所有满足活动最短距离的可回放片段与不可回放片段(相同活动的距离为“0”,反之则为“1”)<sup>[26]</sup>。

**定义 5** 划分可达活动图表。 $R(\wp) = (conc, seq, conf, C(R(\wp)))$  是对可达活动图表所进行的划分,即用并发、序列、冲突 3 种行为关系将其划分为不同的片段(无法被这 3 种行为关系划分的单个活动可被看作一个独立的行为模式)。其中:  $conc$  表示并发片段,即偏差活动之间满足并发关系;  $seq$  表示序列片段,即偏差活动之间满足因果关系;  $conf$  表示冲突片段,即偏差活动之间具有排它性。划分适用于回放过程的可达活动图表需要满足给定的约束条件,即  $C(R(\wp))$ 。 $C(R(\wp))$  中所包含的规则按照优先级排列如下:

- (1)  $p(R(\wp)) = (a_i, seq(\wp)/conc(\wp)/conf(\wp), a_j) \Rightarrow (l(e_0) = a_i/\lambda(t_0) = a_i) \wedge (l(e_n) = a_j/\lambda(t_m) = a_j) \wedge |E| = n \wedge |T| = m$ ;
- (2)  $entry(seq/conc/conf) < (seq/conc/conf) < exit(seq/conc/conf)$ ;
- (3)  $|entry(seq/conc/conf)| = |exit(seq/conc/conf)|$ ;
- (4)  $(\sum_{i=1}^{|seq|} a_i = S \Rightarrow |S| > 1) / (\sum_{i=1}^{|conc|} a_i = P \Rightarrow |P| > 1) / (\sum_{i=1}^{|conf|} a_i = E \Rightarrow |E| > 1) / \sum_{i=1}^{|l(E)/\lambda(T)|} a_i = A \Rightarrow |A| = 1$ ;
- (5)  $C(R(\wp_N)) \approx C(R(\wp_L))$ ;
- (6)  $pre(seq) > pre(conc, conf)$ 。

在上述约束条件中  $pre(seq) > pre(conc, conf)$  表

示优先划分序列关系的片段;  $C(R(\wp_N^*)) \approx C(R(\wp_L))$  表示对过程模型的可达活动图表进行划分必须满足与事件日志中相对应片段之间活动距离最短的规则(相同活动之间的距离为“0”,反之为“1”)。图 1b 和图 1c 划分图 1a 中初始网系统  $N^*$  与事件日志  $L = \{(ABCFKLMO), (ABDEFKML O), (AGHIJKLMO)\}$  的可达活动图表,图中分别用橘色虚线、紫色虚线、蓝色虚线来划分并发片段、序列片段、冲突模式,绿色虚线划分事件日志或者网系统  $N^*$  中所特有的单个活动,而红色虚线部分则用来表示起始活动和终止活动。

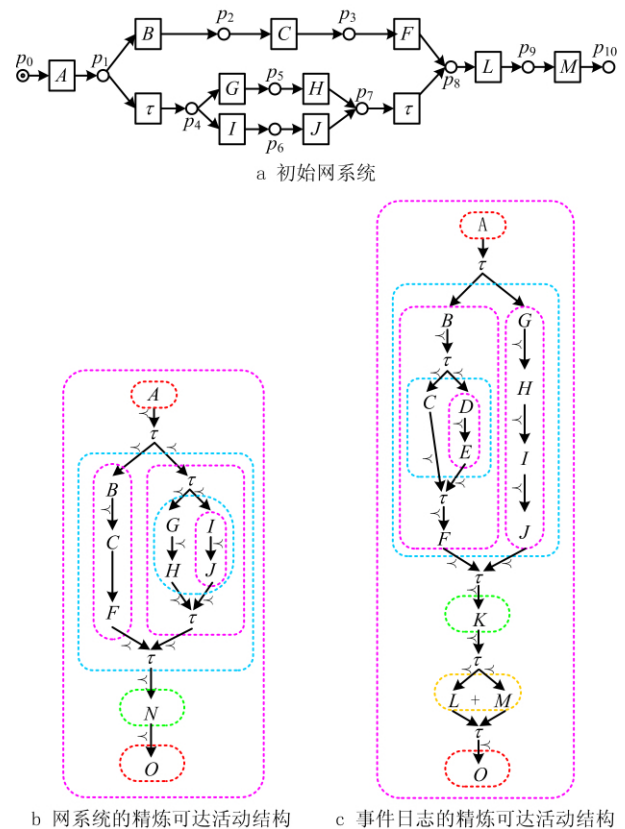


图1 无循环的精炼可达活动结构

根据划分后的可达活动图表可获得事件日志  $L$  与初始网系统  $N^*$  之间满足活动最短距离的对应片段,并记录于表 2。

表 2  $L$  与  $N^*$  的片段

$R(\wp_L)$	A	B	$conf(seq(Bconf(Cseq(DE))F), seq(GHIJ))$	$B_L(K)$	$B_{L+}(LM)$	$\emptyset$	O
$R(\wp_{N^*})$	A	B	$conf(seq(BCF), conf(seq(GH), seq(IJ)))$	$\emptyset$	$\emptyset$	$B_{N^+}(N)$	O

其中灰色阴影部分表示 4 对不可回放的片段,且第一组片段中包含若干个子片段。

## 2.2 确定偏差的发生、位置以及行为关系

最优对齐是指事件日志与过程模型之间拥有最

低成本的对齐,即  $\text{cost}(\xi_{op}) \leq \text{cost}(\xi)$ , 且  $\forall \xi \in \xi^*$ ,  $\exists \xi_{op} \in \xi^*$ 。为了更清楚地阐述如何构建偏差子结构,首先将图 1 中划分的所有对应片段转换为行为模式。并记录于表 3。

表 3  $L$  与  $N^*$  的行为模式

$B_L$	A	B	$B_L \times (CB_L \prec (DE))$	F	$B_L \prec (GHIJ)$	$B_L(K)$	$B_L^+(LM)$	$\emptyset$	O
$B_{N^*}$	A	B	$B_{N^*}(C)$	F	$B_{N^*} \times (B_{N^*} \prec (GH) B_{N^*} \prec (IJ))$	$\emptyset$	$\emptyset$	$B_{N^*}(N)$	O

通过行为关系的一致性校验,可回放的行为模式被直接合并为两个对应模式之间的一致行为,表

4 中不可回放的行为模式根据定义 4 中的规定以及相应模式之间的差异行为进行合并。

表 4  $L$  和  $N^*$  之间的可回放与不可回放行为模式

$B_{L/N^*}$	A	B	$B_{L \subseteq^*} \times (B_{N^*}(C) / B_{L \subseteq} \prec (DE))$	F	$B_{L \subseteq} \times (B_{L \subseteq} \prec (GH) B_{L \subseteq} \prec (IJ))$	$B_{L(e)}(K)$	$B_{L \subseteq^+}(LM)$	$B_{\lambda(t)}(N)$	O
-------------	---	---	--	---	--	---------------	-------------------------	---------------------	---

由此可获得事件日志与过程模型之间的两组最优对齐  $\xi_{op1}$  并记录于表 5。

表 5 最优对齐

$\xi_{op1}$	A	B	→	D	E	F	G	H	I	J	K	L	M	→	O
	A	B	C	→	→	F	G	H	→	→	→	→	→	N	O
$\xi_{op2}$	A	B	→	D	E	F	G	H	I	J	K	M	L	→	O
	A	B	C	→	→	F	G	H	→	→	→	→	→	N	O

表中背景颜色用来区分不同行为模式所产生的偏差,由于  $B_{L \subseteq^*} \times (B_{N^*}(C) / B_{L \subseteq} \prec (DE))$  可拆分为  $B_{\lambda(t)}(C)$  和  $B_{L \subseteq} \prec (DE)$ ,将最优对齐检测出的偏差分为以下五组:  $\{EsertC\}$ 、 $\{EsertD/EsertE\}$ 、 $\{EsertI, EsertJ\}$ 、 $\{EsertK\}$ 、 $\{EsertL, EsertM\}$ 、 $\{SkipN\}$ 。根据偏差种类的不同,使用以下两种方法分别确定它们的修复位置:①插入偏差的位置是由最优对齐中产生该偏差的非同步移动前最后一个同步移动中的变迁在网系统  $N^*$  上的后置库所来确定的,记作  $\text{loc}(EsertI(e)) = (\sigma(a'))^*$ ,  $\lambda((\sigma(a'))^*) = a$ ;②跳过偏差是将网系统  $N^*$  上移动变迁的前置库所和后置库所分别作为隐变迁  $\tau$  的起始点和终止点,记作  $\text{loc}(Skip\lambda(t)) = (\sigma(a)) \cup (\sigma(a'))^*$  ( $\sigma$  是映射函数,将活动标签转换为网系统  $N^*$  上的变迁)。根据不可回放行为模式的划分,可从上述最优对齐  $\xi_{op}$  中检测到 3 组直接跟随的偏差:  $\overline{M}_{L1D-F} = \{D, E\}$ 、 $\overline{M}_{L2D-F} = \{I, J\}$ 、 $\overline{M}_{N^*1D-F} = \{L, M\}$ ,并基于它们所对应的不可回放行为模式判定每组偏差元素之间的行为关系分别为:序列、序列、并发。

通过确定可回放与不可回放的行为模式产生最

优对齐,且依靠行为关系的一致性校验检测偏差元素的完整信息。由于事件日志与过程模型之间所有可回放的行为模式无需比较,仅搜索不可回放行为模式的最优对齐。由于偏差的完整信息被看作是模型修复的输入信息,这种方法能够减少最优对齐搜索的工作量且不影响模型修复的效率。

## 3 关于事件日志对过程模型所做的修复

本章介绍一种方法,该方法基于行为关系将事件日志与过程模型之间的直接跟随偏差构建为子结构,从而通过减少单次发生的插入偏差个数来提升精度。根据各种不可回放行为模式所产生的不同偏差行为,可对其修复进行具体的分析。

### 3.1 基于偏差子结构的修复操作

通过构建直接跟随偏差的子结构过程模型上的修复主要依赖于并发、序列、冲突 3 种行为关系。由于具有排他性的偏差元素之间不存在直接跟随关系,可以将冲突结构中每条分支上的偏差元素根据并发或序列关系构建为子结构,并分别以自循环的方式插入过程模型中合适的位置。各种不可回放行为模式中直接跟随偏差所组合的子结构类型可被概括为以下两种:  $N_{d-E(l(e), l(e'))^*} / N_{S(\lambda(t), \lambda(t))^-}$ ,  $N_{d-E(l(e), l(e'))^+} / N_{S(\lambda(t), \lambda(t))^+}$ 。图 2a 和图 2b 描述了上述两种子结构的具体操作,它们都是以自循环的方式插入到网系统  $N^*$  中的。在实际修复过程中,如果一个不可回放行为模式中包含若干个子模式,则可能导致直接跟随偏差的并发子结构嵌套在序列子结构中或者序列子结构嵌套于并发子结构中。如图 2c 和图 2d 所示,使用 2.2 节所介绍的方法对



偏差完整信息进行检测,并将这两种特殊的偏差子结构记作:  $N_{d-E(l(e),l(e'))_+,l(e'')_+}/N_{S(\lambda(t),\lambda(t'))_+,\lambda(t'')_+}$ ,  $N_{d-E(l(e),l(e'))_+,l(e'')_+}/N_{S(\lambda(t),\lambda(t'))_+,\lambda(t'')_+}$ 。值得注意的,当一个不可回放行为模式中包含的偏差或者子结构都来自于过程模型时,即只会通过跳过偏差行

为对该模式进行修复,则修复后过程模型与事件日志之间的适合度能够得到改善但对精度的提升没有影响,如:  $B_{(\times/+)(\times/ < )}$ ,  $B_{N^* \subseteq (+, <, \times)}$ ,  $B_{\lambda(t)}$ 。因此,由跳过偏差组成的子结构不能对精度的提升产生帮助。

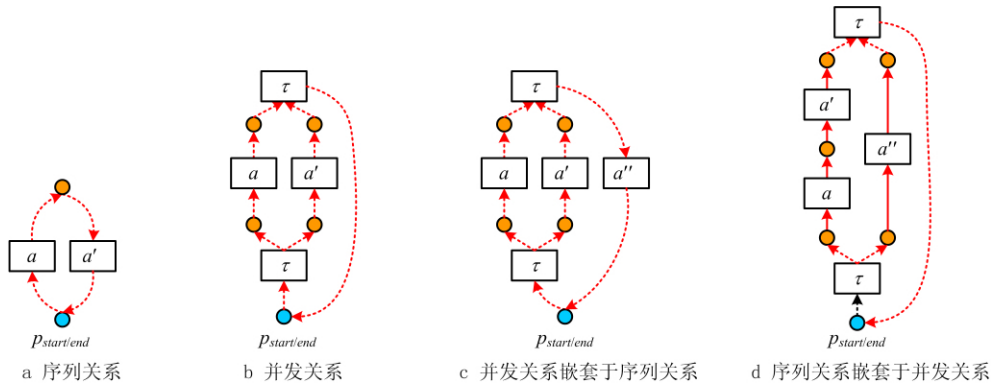


图2 构建各种行为关系的偏差子结构

图 2 中的方法可以将若干个偏差活动构建为一个子结构,从而将所有直接跟随的偏差活动看作 1 个。事件日志与过程模型之间拟合度的度量公式为

$$Fitness = 1 - \frac{Ccost}{|L| + |M_L|},$$

其中:  $Ccost$  表示业务流程中的所有偏差成本,  $|L| + |M_L|$  为事件日志的尺寸以及与其进行最优对齐的模型尺寸。本文方法以保证适合度为前提,因此统一使用自循环插入方式修复日志中的可观测偏差。事件日志与过程模型之间精度的度量公式为

$$Precision = \frac{|L| - Ecost}{|L|},$$

其中  $Ecost$  是指插入偏差的总成本。准确修复表示偏差活动在过程模型与事件日志中发生的形式完全一致,而使用自循环方式修复后每个单次插入偏差的回放准确度为  $1/\infty = 0$ 。因此,自循环插入方式只能修复迭代的插入偏差。将所有直接跟随的单次发生偏差根据特定的行为关系构建为子结构,则可以通过减少事件日志的尺寸提升精度。例如,直接跟随的  $n$  个单次发生偏差成本为  $Ecost'$ ,将它们构建为一个子结构后精度为

$$Precision = \frac{|L| - Ecost}{|L| - Ecost' \times ((n-1)/n)}.$$

值得注意的是,这种构建子结构的方式仅针对单次发生的插入偏差,而不适用于由日志中自循环产生

的迭代插入偏差。迭代插入偏差需要使用自循环的方式单个插入过程模型,以确保正常的回放。

### 3.2 各种不可回放行为模式的修复

因为本节所提方法的修复目的是为了有效改善事件日志与过程模型之间的精度,所以主要介绍如何修复包含插入偏差的不可回放模式,即并发/因果模式并发/冲突模式。因果/冲突模式,以及事件日志的行为包含模式。

#### 3.2.1 行为轮廓所产生差异的修复

行为轮廓主要指因果、并发、序列 3 种行为关系,本节对其中任意两种行为之间产生的差异进行修复。并发/因果模式是并发关系中的部分行为包含模式,它与并发/冲突模式分别表示事件日志与过程模型相对应模式中活动之间的行为关系为并发与因果或者并发与冲突。由于并发模式中活动发生顺序的任意性,对并发/因果模式和并发/冲突模式进行修复首先需要通过最优对齐检测出偏差的完整信息,并将具有直接跟随关系的偏差元素构建为子结构。因果/冲突模式是指两个相对应模式中活动之间的行为关系分别为因果和冲突,它们所产生的偏差行为在冲突模式中包含活动个数最小的一条分支上。对该模式进行修复可将这条分支上所有偏差元素构建为序列子结构。假设图 3 中初始网系统  $N^*$  与事件日志  $L$  之间的对应行为模式记录于表 6。

表 6 可回放与不可回放行为模式

$B_{L/N^*}$	A	$B_{+/\prec}(B_{\prec/\prec}(BC)/B_{\prec/\prec}(DE))$	$B_{+/\times}(B_{\prec/\prec}(FG)B_{\prec/\prec}(HI))$	$B_{\prec/\times}(B_{\prec/\prec}(JKL)B_{\prec/\prec}(MN))$	O
-------------	---	--	--	---	---

根据最优对齐检测出的偏差完整信息,图 3 中采用构建子结构的方式对上述 3 种不可回放的行为模式进行修复:①红色虚线部分是对插入偏差进行的修复,图 3 中分别用蓝色填充和橘色填充的库所来标注每个插入偏差/偏差子结构在过程模型上的位置以及子结构中的连接库所;②紫色虚线部分是对跳过偏差的修复,其中绿色填充的库所用来表示插入偏差与跳过偏差的共同位置。修复并发/因果模式中产生的直接跟随跳过偏差  $\{D,E\}$  不会对精度造成影响,这里根据行为关系将它们构建为子结构只能在一定程度上降低修复后过程模型的复杂度<sup>[27]</sup>。如图 3 所示,并发/因果模式与并发/冲突模式产生的差异既包括单个偏差活动也包括偏差子结构,而因果/冲突模式的差异则只有序列子结构。由于同一个位置的相同偏差重复出现多次成本仍然为

1,因此从提升精度的角度考虑,可将  $Esert(l(e)) \cup \{Esert(l(e)), Esert(l(e'))\} = \{Esert(l(e)), Esert(l(e'))\}$  看作是  $Esert(l(e)) \cup N_{E(l(e),l(e'))\prec}$ 。图 3 基于  $\bar{M}_{L1D-F}^{\prec} = \{D,E\}$ 、 $\bar{M}_{L2D-F}^{\prec} = \{F,G\}$ 、 $\bar{M}_{L3D-F}^{\prec} = \{M,N\}$  三组直接跟随偏差之间的序列关系构建偏差子结构进行修复,即:  $N_{d-E(D,E)\prec}$ 、 $N_{d-E(F,G)\prec}$ 、 $N_{d-E(M,N)\prec}$ 。

3.2.2 行为包含模式的修复

事件日志的行为包含模式是指在当前模式中过程模型可以捕获的行为都被记录于事件日志中,但事件日志中可观测的行为则不能完全在过程模型中被捕获<sup>[28]</sup>。根据定义 4 中行为包含模式的各种类型,假设图 4 中初始网系统  $N^*$  与事件日志  $L$  之间的对应行为模式被记录于表 7。

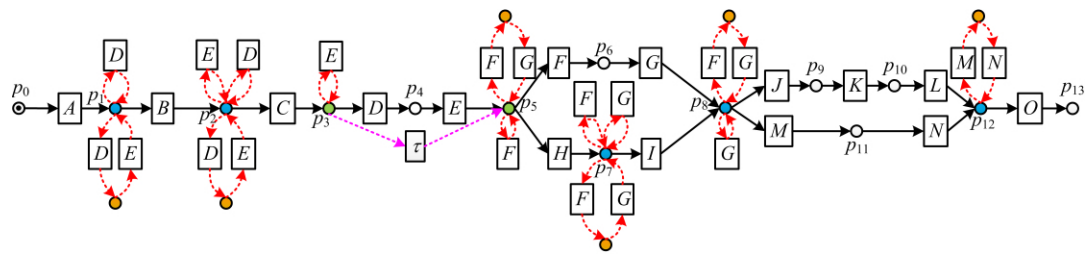


图3 基于并发/因果、并发/冲突以及因果/冲突3种模式修复后的网系统 $N^*$

表 7 可回放与不可回放行为模式

$B_{L/N^*}$	A	$B_{\times/\times}(B_{L\subseteq\cdot\cdot}(B_N\cdot(B)/B_{L\times}(BB_{L\subseteq\prec}(CD))))$ , $B_{\prec/\prec}(EF)$	$B_{L\subseteq+}(GH)$	$B_{L\subseteq\times}(IB_{L\subseteq\prec}(JK))$	$B_{L\subseteq\prec}(LMN)$	O
-------------	---	--	-----------------------	--	----------------------------	---

根据最优对齐检测出的偏差完整信息,图 4 使用红色虚线和紫色虚线分别描述了对插入偏差和跳过偏差的修复。 $B_{L\subseteq\cdot\cdot}(B_N\cdot(B)/B_{L\times}(BB_{L\subseteq\prec}(CD)))$  的修复可被看作是  $B_{\lambda(l)}(B)$  和  $B_{L\subseteq\prec}(CD)$  两个模式,它所产生的偏差分别为:  $Skip(B)$ ,  $\{Esert(C), Esert(D)\}$

两部分。其他 3 个事件日志的整体行为包含模式中只包含插入偏差,且它们的修复操作是根据当前模式中包含的所有偏差及偏差之间的行为关系所构建的。因此,图 4 根据 4 组直接跟随偏差  $\bar{M}_{L2D-F}^{\prec} = \{C,D\}$ 、

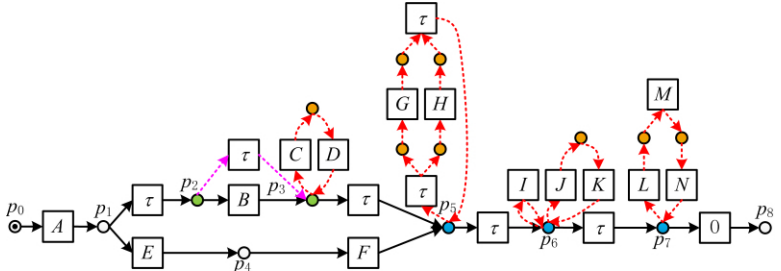


图4 基于行为包含模式修复后的网系统 $N^*$

$\mathcal{M}_{L2D-F} = \{G, H\}$ 、 $\mathcal{M}_{L2D-F} = \{J, K\}$ 、 $\mathcal{M}_{L2D-F} = \{L, M, N\}$  之间的行为关系构建偏差子结构对过程模型进行修复, 即  $N_{s-E(C,D)}<$ ,  $N_{s-E(G,H)}>$ ,  $N_{s-E(J,K)}<$ ,  $N_{s-E(L,M,N)}<$ 。这里, 偏差  $\{J, K\}$  是整体冲突行为包含模式中一条分支上的活动, 它与另一条分支上的偏差活动  $\{I\}$  之间具有排他性, 因此将  $\{J, K\}$  与  $\{I\}$  分别插入过程模型的同一个位置。

算法 1 描述了各种不适合行为模式的修复操作, 首先将空集赋予修复操作的集合  $R_0$ , 然后主要对以下 2 部分进行分析:

(1) 行为轮廓所产生差异的修复操作主要分为以下 3 种情况: ① 并发/因果模式。当该模式中的活动总数不超过 3 时, 修复操作插入并且跳过后一个活动; 反之, 其操作是插入并且跳过后半条分支上所有活动的子结构(第 5~第 7 行)。② 并发/冲突模式, 这种模式的修复操作类似于并发/因果, 但它没有跳过偏差, 且插入的活动或者子结构来自于两条分支中的任意一条(第 8~10 行)。③ 因果/冲突模式。该模式的修复操作仅将包含活动个数少的一条分支上的偏差行为插入过程模型, 其可能是单个活动或者子结构(第 11~第 13 行)。

(2) 因果、冲突、并发关系所产生的整体行为包含模式的修复操作, 以及冲突模式中部分行为包含的修复操作。整体行为包含可根据偏差之间的行为关系构建该模式中所有偏差元素的子结构(第 14~第 16 行)。事件日志的冲突行为部分包含模式需要分别插入和跳过它与过程模型之间不相交以及相交行为中的单个活动或者直接跟随偏差的子结构(第 17~19 行)。

算法 1 基于偏差于结构的修复操作。

输入: net  $N$ , Log  $L$ ,  $R(\mathcal{P}_L)$ ,  $R(\mathcal{P}_N)$ , icost;

输出:  $R_0$ 。

```

1:  $R_0 \leftarrow \emptyset$ 
2: for each  $e_i \in L$  do
3:   for each  $t_j \in T$  do
4:     for each  $a_k \in \mathbb{S} \cup \mathbb{Z}$  do
5:       if  $l(e_i) = \lambda(t_j) = a_k \wedge B_{L+}((a_1, a_2) / ((a_1, \dots, a_n) <, (a_{n+1}, \dots, a_{n+m})_L)) \wedge B_N^* < ((a_1, a_2) / ((a_1, \dots, a_n) <, (a_{n+1}, \dots, a_{n+m})_L))$  then
6:          $R_0 \leftarrow (SEsert(a_1) \cup Skip(a_1)) / (N_{sl-E(a_{n+1}, \dots, a_{n+m})} < \cup SEsert(a_{n+1}, \dots, a_{n+m}) \cup N_{S(a_{n+1}, \dots, a_{n+m})} <)$ 
7:       end
8:       if  $l(e_i) = \lambda(t_j) = a_k \wedge B_{L+} \times ((a_1, a_2) / ((a_1, \dots, a_n) <, (a_{n+1}, \dots, a_{n+m}) <)) \wedge B_N^* \times / < ((a_1, a_2) / ((a_{n+1}, \dots, a_{n+m}) <))$  then
9:          $R_0 \leftarrow (SEsert(a_1, a_2) / (N_{sl-E(a_1, \dots, a_n)} < / (a_{n+1}, \dots, a_{n+m}) \cup SEsert(a_1, \dots, a_n, a_{n+1}, \dots, a_{n+m})) / ((skip(a_1) \cup skip(a_2)) / N_{S(a_1, \dots, a_n)} < / (a_{n+1}, \dots, a_{n+m}) <)$ 

```

```

10:   end
11:   if  $l(e_i) = \lambda(t_j) = a_k \wedge B_{L+} \times ((a_1, a_2) / ((a_1, \dots, a_n) <, (a_{n+1}, \dots, a_{n+m}) <)) \wedge B_N^* \times / < ((a_1, a_2) / ((a_1, \dots, a_n) <, (a_{n+1}, \dots, a_{n+m}) <))$  then
12:      $R_0 \leftarrow (SEsert(a_1/a_2) / N_{sl-E(a_1, \dots, a_n)} < / (a_{n+1}, \dots, a_{n+m}) <) / (skip(a_1/a_2) / N_{S(a_1, \dots, a_n)} < / (a_{n+1}, \dots, a_{n+m}) <)$ 
13:   end
14:   if  $l(e_i) = \lambda(t_j) = a_k \wedge B_{+} \times ((a_1, a_2) / ((a_1, \dots, a_n) <, (a_{n+1}, \dots, a_{n+m}) <)) \in R(\mathcal{P}_L/N^*)$  then
15:      $R_0 \leftarrow N_{sl-E(a_1, a_2) / ((a_1, \dots, a_n), (a_{n+1}, \dots, a_{n+m}))} + / < \times / N_{S((a_1, a_2) / ((a_1, \dots, a_n), (a_{n+1}, \dots, a_{n+m})))} + / < \times$ 
16:   end
17:   if  $B_{L+} \times ((a_1, a_2) / ((a_1, \dots, a_n), (a_{n+1}, \dots, a_{n+m}))) \in R(\mathcal{P}_L) \wedge B_N^* (a_1 / (a_1 \dots a_n)) \in R(\mathcal{P}_N^*)$  then
18:      $R_0 \leftarrow (SEsert(a_2) \cup Skip(a_1)) / (N_{sl-E(a_{n+1}, \dots, a_{n+m})} < \cup N_{S(a_1, \dots, a_n)} <)$ 
19:   end
20:   end
21: end
22: end
23: return  $R_0$ 

```

## 4 评估

本章记录了本文方法与当前两种方法之间修复性能的比较结果<sup>[13]</sup>。本文方法在实验中主要根据行为关系将直接跟随偏差元素构建为子结构, 并使用自循环插入的方式进行修复。

### 4.1 实验设置

#### 4.1.1 工具集

当前, 模型修复可以使用 Prom 框架进行验证, 但其原始代码并没有提供本文方法中的改进部分。为保证实验结果的公平性, 本文在此基础上使用 Java 语言编写了插件 M-repair, 其可以根据不同方法的性能分别校验修复后数据集的适合度与精度。实验使用 Java 语言编写类似于 CPN Tools 功能的插件 PSLG, 其利用活动的随机触发性自动生成常见格式或 xes 格式的事件日志。

#### 4.1.2 数据集

实验使用 4 个数据集进行评估, 它们分别来自人造业务流程和真实业务流程。因为每个业务流程中都包含一个无循环结构的数据集和一个有循环结构的数据集, 所以分别记作:  $D_a$ ,  $D_{la}$ ,  $D_r$ ,  $D_{lr}$ , 其中真实业务流程来自于中国一家人寿保险公司的投保流程。通过模拟 4 个业务流程的发生实例分别获得与它们一致的 4 个序列集  $\delta^*$ , 并将数据集中存在的不可回放片段化分为以下 3 种: ① 并发\因果\冲突行为以及单个活动插入/跳过产生的不一致行为; ② 行为包含产生的不一致行为; ③ 循环路径产生的不行为。为了指导实验的正常进行可根据上述 3 种



情况下的不适合行为模式对发生序列进行更改,产生与业务流程不一致的执行序列集 $\sigma^*$ ,每个执行序列集与其对应的发生序列集组成一个数据集。根据执行序列集 $\sigma^*$ 中包含的所有活动及其之间的行为构建过程模型,并使用 PSLG 插件将其自动生成事件日志。因为 PSLG 插件所生成的事件日志包含 500 多条不同案例的迹,所以选择可以反映所有活动及其之间行为关系的 10 条不同案例的迹,并在每条迹的发生次数之上模拟 20 次。4 个数据集中

的事件日志包含 200~360 条迹,且它们的尺寸分别为  $L^* = \{5\,920, 5\,860, 6\,160, 8\,880\}$ 。

4.2 适合度确定为 1 的实验评估

为保证适合度在修复后始终能够达到 1,本节使用的现存修复方式与所提方法都是以自循环的方式修复插入偏差。两种方法在适合度与精度上的度量可参照表 8 中记录的公式,且它们之间精度的差异来自于构建偏差子结构对日志尺寸的减少。

表 8 修复性能的度量公式

插件	适合度	精度
Repair-check	$1 - \frac{Ccost - Rcost}{ L  +  M_L }$	$1 - \frac{Ccost - Ecost + Icost + cost_{loop}}{ L }$
Mrepair-check*	$1 - \frac{Ccost - Rcost}{ L  +  M_L }$	$1 - \frac{Ccost - Ecost + Icost + cost_{loop}}{ L  - (Rcost - Icost) \times ((n-1)/n)}$

4.2.1 在含有迭代插入偏差数据集上的评估

图 5a 和图 5b 中分别描述了来自真实业务流程数据集的精度值比较,其中非循环部分都存在差异。图 5a 中的数据集没有来自于循环部分的差异,两种方法在精度上的提升仅依靠于对迭代可观测偏差的准确修复(迭代可观测偏差产生于事件日志中的自

循环路径)。图 5b 的数据集中事件日志的一个模式有循环路径,而过程模型中与之相对应的模式则没有。由于该循环部分的不可回放模式同时由它们的非循环部分和返回路径所产生,它们统一按照非循环部分的方法进行修复。由此可知,两种方法在循环部分与非循环部分上就精度改善而言的差异是一

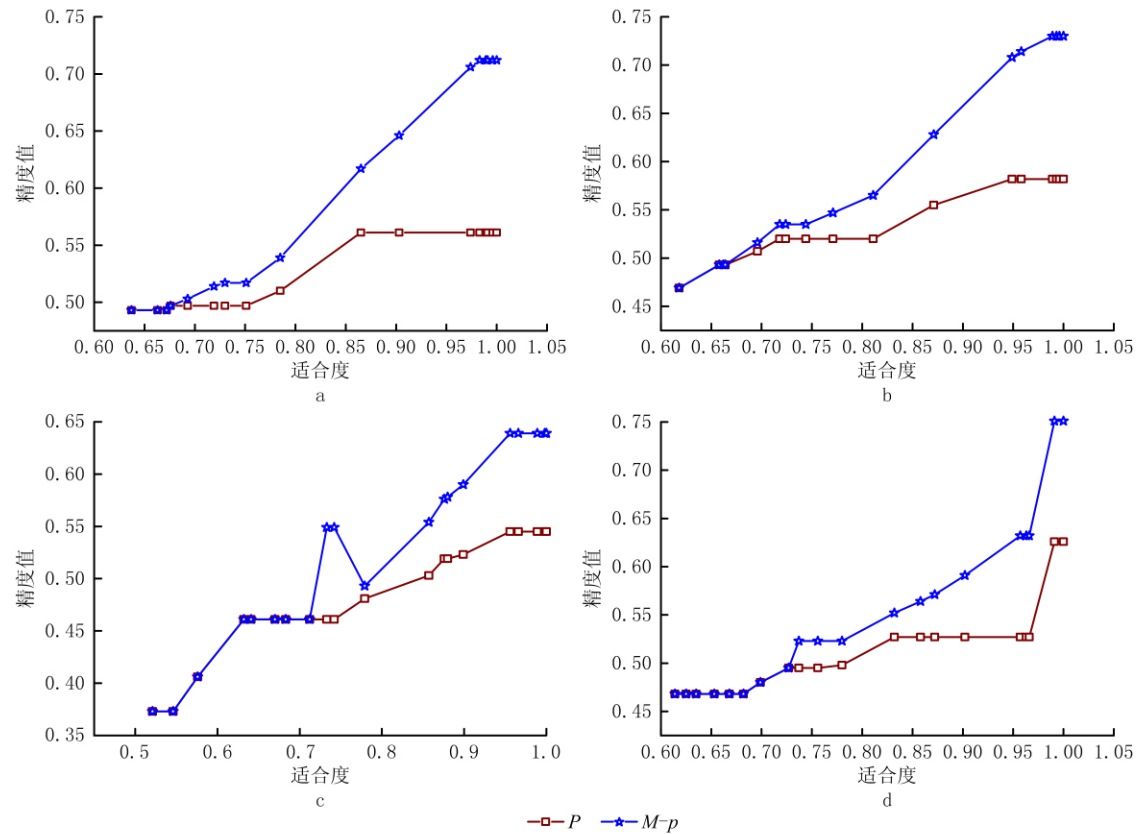


图5 带有迭代插入偏差的不同数据集上精度值的比较结果

样的。图 5c 和图 5d 中分别描述了来自人造业务流程数据集的精度值比较,其中非循环部分都存在差异。图 5c 只在非循环部分产生差异,因此两种方法对精度的改善与图 5a 一样。图 5d 的数据集中事件日志的一个模式有循环路径。由于该循环部分的不可回放模式仅由事件日志中包含的循环路径所产生,返回路径上没有活动,使用本文所提出的方法和现存方法都可以准确地修复由它产生的偏差。从图中 4 组数据可以看出,相较于现存方法而言,本文方法可以在保证适合度的基础上平均提升精度约 13%,其中  $P$  与  $M-p$  分别表示现存方

法与本文方法所产生的精度值。

#### 4.2.2 在不含有迭代插入偏差数据集上的评估

图 6 中的 4 个数据集是在图 5 所使用的数据集基础上进行的改进,即其中 4 个事件日志都不包含迭代可观测偏差。由于仅使用自循环方式插入可观测偏差无法提升精度,现存方法在图 6a~图 6c 中都始终保持初始的值。图 6d 中现存方法的精度值增长依赖于循环所引起的重复活动。本文方法在图 6a~图 6c 中的精度增长点都依赖于将该模式中直接跟随插入偏差构建为子结构的修复,而图 6d 中的精度增长还有循环所产生偏差的修复。

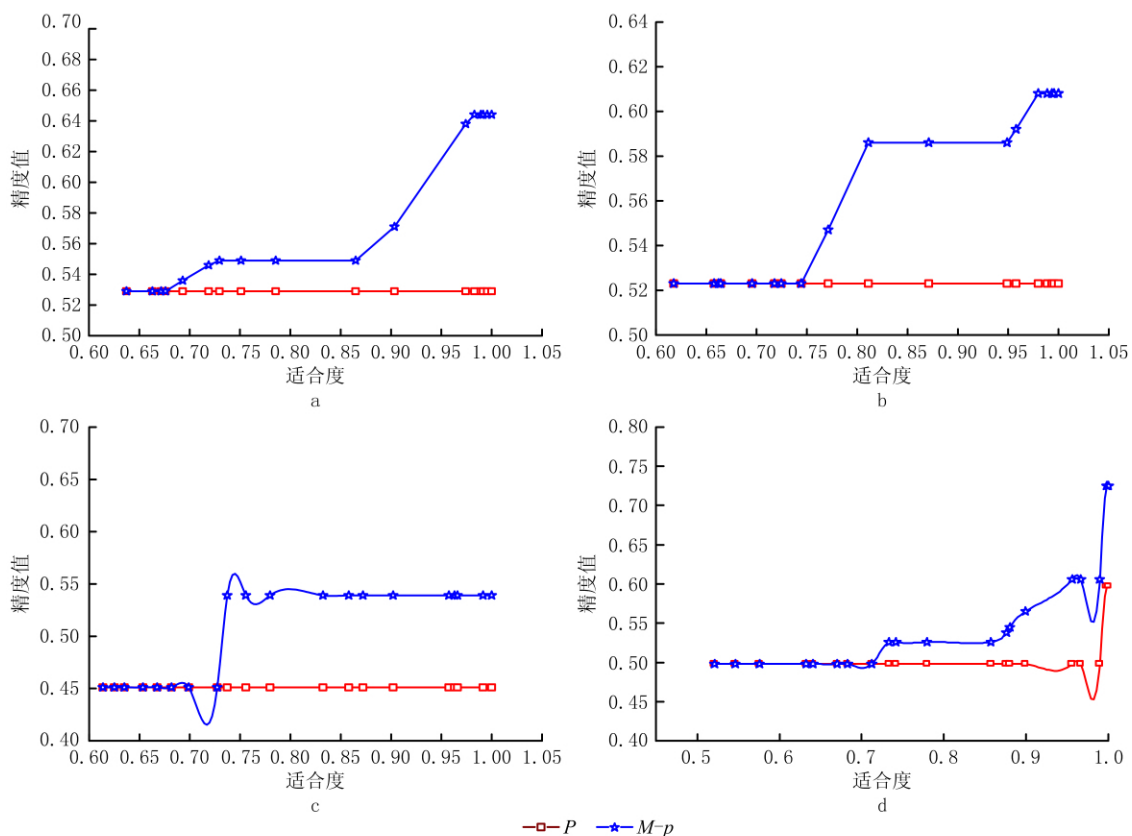


图6 没有迭代插入偏差的不同数据集上精度值的比较结果

## 5 结束语

本文提出一种新的有效方法来修复事件日志无法回放于过程模型的行为,该方法主要解决了当前模型修复方法在精度问题上的局限性。首先,在给定约束条件下划分事件日志与过程模型的可达活动图表,并由此产生它们之间的对应行为模式;然后,通过行为关系的一致性校验将对应行为模式合并为可回放与不可回放的模式并产生最优对齐。根据这种最优对齐所检测出的偏差行为关系将所有直接跟

随的偏差元素构建为子结构;最终,采用自循环插入/跳过式的方法对偏差元素/偏差子结构进行修复。本文所提方法能够通过减少日志所产生的插入偏差,实现在保证适合度的前提下尽可能的提升精度。

本文方法是针对静态过程模型进行的实时修复,并没有考虑当需求与环境不断变化时过程模型在实际运行中可能产生的额外行为。未来工作将分析这种修复方式如何解决动态过程模型与事件日志之间的修复问题。

## 参考文献:

- [1] VAN DER AALST W M P. Process mining: Discovery, conformance and enhancement of business processes[EB/OL]. [2020-05-16]. <https://link.springer.com/content/pdf/bfm%3A978-3-642-19345-3%2F1.pdf>.
- [2] VAN DER AALST W, WEIJTERS T, MARUSTER L. Workflow mining: Discovering process models from event logs[J]. IEEE Transactions on Knowledge and Data Engineering, 2004, 16(9):1128-1142.
- [3] VAN DER AALST W M P. Distributed process discovery and conformance checking[J]. Lecture Notes in Computer Science, 2012, 7212: 1-25.
- [4] WEIDLICH M, POLYVYANY A, DESAI N, et al. Process compliance measurement based on behavioral probes[J]. Lecture Notes in Computer Science, 2010, 6051: 499-514.
- [5] BUIJS J C A M, VANDONGEN B F, VANDER AALST W M P. On the role of fitness, precision, generalization and simplicity in process discovery[J]. Lecture Notes in Computer Science, 2012, 7565: 305-322.
- [6] ROZINAT A, VANDER AALST W M P. Conformance checking of processes based on monitoring real behavior[J]. Information Systems, 2008, 33(1): 64-95.
- [7] JAGADEESH CHANDRA BOSE R P, VANDER AALST W M P. Trace alignment in process mining: Opportunities for process diagnostics[J]. Lecture Notes in Computer Science, 2010, 6336: 227-242.
- [8] VAN DER AALST W M P, RUBIN V, VERBEEK H M W, et al. Process mining: A two-step approach to balance between under-fitting and over-fitting[J]. Software and Systems Modeling, 2010, 9(1): 87-111.
- [9] MACEDO N, JORGE T, CUNHA A. A feature-based classification of model repair approaches[J]. IEEE Transactions on Software Engineering, 2017, 43(7): 615-640.
- [10] VAN DER AALST W M P, ADRIANSYAH A A, VANDONGEN B F. Replaying history on process models for conformance checking and performance analysis[J]. Wiley Interdisciplinary Reviews—Data Mining and Knowledge Discovery, 2012, 2(2):182-192.
- [11] POLYVYANY A, VANDER AALST W M P, TERHOFSTEDE A H M, et al. Impact-driven process model repair[J]. ACM Transactions on Software Engineering and Methodology, 2017, 25(4):1-60.
- [12] FAHLAND D, VANDER AALST W M P. Model repair—Aligning process models to reality[J]. Information Systems, 2015, 47:220-243.
- [13] VANDENBROUCKE S K L M, DEWEERDT J, VAN THIENEN J, et al. Determining process model precision and generalization with weighted artificial negative events[J]. IEEE Transactions on Knowledge and Data Engineering, 2014, 26(8):1877-1889.
- [14] ADRIANSYAH A, MUNOZGAMA J, CARMONA J, et al. Measuring precision of modeled behavior[J]. Information Systems and E-Business Management, 2015, 13(1):37-67.
- [15] VAN DER AALST W M P. Process discovery from event data: relating models and logs through abstractions[J]. Wiley Interdisciplinary Reviews—Data Mining and Knowledge Discovery, 2018, 8(3):1-21.
- [16] SONG Wei, XIA Xiaoxu, JACOBSEN H A, et al. Efficient alignment between event logs and process models[J]. IEEE Transactions on Services Computing, 2017, 10(1):136-149.
- [17] GARCIBANUELOS L, VANBEEST N R T P, DUMAS M, et al. Complete and interpretable conformance checking of business processes[J]. IEEE Transactions on Software Engineering, 2017, 44(3):262-290.
- [18] GRONER G, BOSKOVIC M, PARREIRAS F S, et al. Modeling and validation of business process families[J]. Information Systems, 2013, 38(5):709-726.
- [19] FRANCESCO MARINO C D, TIELLA R, GHIDINI C, et al. A multi-objective approach to business process repair[J]. Lecture Notes in Computer Science, 2014, 8831:32-36.
- [20] CERVANTES A A, VAN BEEST N R T P, ROSA M L, et al. Interactive and incremental business process model repair[J]. Lecture Notes in Computer Science, 2017, 10573: 53-74.
- [21] QI Hongda, DU Yuyue, QI Liang. An approach to repair Petri net-based process models with choice structures[J]. Enterprise Information Systems, 2018, 12(8-9):1149-1179.
- [22] ZHANG Xize, DU Yuyue, QI Lang, et al. Repairing process models containing choice structures via logic Petri nets[J]. IEEE Access, 2018, 6:53796-53810.
- [23] TIAN Yinhua, DU Yuyue, HAN Dong, et al. Aligning event logs and process models based on Petri nets[J]. Computer Integrated Manufacturing Systems, 2019, 25(4): 809-829 (in Chinese). [田银花, 杜玉越, 韩冬, 等. 基于Petri网的事件日志与过程模型对齐方法[J]. 计算机集成制造系统, 2019, 25(4):809-829.]
- [24] NIELSEN M, PLOTKIN G D, WINSKEL G. Petri nets, event structures and domains, Part I[J]. Theoretical Computer Science, 1981, 13(1): 85-108.
- [25] ROZINAT A, VANDER AALST W M P. Conformance checking of processes based on monitoring real behavior[J]. Information System, 2008, 33(1):64-95.
- [26] VAN HATALO J, VOZLER H, KOEHLER J. The refined process structure tree[J]. Data and Knowledge Engineering, 2009, 68(9): 793-818.
- [27] FAHLAND D, VANDER AALST W M P. Simplifying discovered process models in a controlled manner[J]. Information Systems, 2013, 38(4):585-605.
- [28] FANG Xianwen, ZHAO Fang, FANG Huan, et al. The fusion analysis method about the change region of the business process model based on Behavior Inclusion in Petri net[J]. Chinese Journal of Computers, 2018, 41(3): 695-706 (in Chinese). [方贤文, 赵芳, 方欢, 等. 基于Petri网Behavior Inclusion的业务流程变化域融合分析[J]. 计算机学报, 2018, 41(3): 695-706.]

## 作者简介:

张力雯(1988—),女,安徽淮南人,博士研究生,研究方向:Petri网、配置优化及模型修复,E-mail:1758678508@qq.com;  
 方贤文(1975—),男,河南信阳人,教授,博士生导师,博士,研究方向:Petri网和可信软件,通讯作者,E-mail:280060673@qq.com;  
 邵叱风(1995—),男,安徽合肥人,硕士研究生,研究方向:Petri网和过程挖掘,E-mail:fc\_shao@126.com;  
 王丽丽(1982—),女,安徽安庆人,副教授,博士研究生,研究方向:Petri网、业务流程挖掘等,E-mail:wanglili198301@163.com。