

# 业务流程一致性检查方法

## ——基于直接跟随规则模型

左殷恺<sup>1,2</sup>, 卢可<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. 安徽省煤矿安全大数据分析和预警技术工程实验室, 安徽 淮南 232001

**摘要:**目的 针对传统一致性检测方法大多只考虑控制流层面,并未考虑流程中数据对于一致性检查过程的影响这一局限性,提出一种基于直接跟随规则模型(DFRM)的一致性检查方法,扩展了传统的直接跟随模型。方法 首先将现实流程模型的控制流和数据流分别转换为各自视角的业务流程规则,然后利用关联规则将控制流规则与数据流规则相结合,并利用满足性模理论(SMT)对数据规则进行处理,最终实现基于直接跟随规则模型的表现形式;接着通过从事件数据中提取事件日志,优先对控制流对齐,然后再进行数据流对齐,实现多视角业务流程一致性检查。结果 利用真实医疗事件日志对所提出的方法进行评估,与其他传统方法相比,所提出的方法在模型的拟合度、F1 分数上都有较好表现且直接跟随规则模型,因其模型简单,具有较少的执行时间。结论 将数据视角以数据关联规则的形式与直接跟随模型相结合,这为目前业务流程商业软件的模型表现形式提供了新的扩展方向;将数据视角引入到业务流程中,解决了现有流程模型无法表达与数据相关的一些决策需求问题,对一致性检查过程中其他视角的可能偏差也纳入了考量。

**关键词:**一致性检测;关联规则;智慧医疗;直接跟随规则模型(DFRM)

**中图分类号:**TP391.9 **文献标识码:**A **doi:**10.16055/j.issn.1672-058X.2025.0002.013

### A Conformance Checking Method for Business Process: Based on Directly Following Rule Model

ZUO Yinkai<sup>1,2</sup>, LU Ke<sup>1,2</sup>

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

2. Anhui Province Engineering Laboratory for Big Data Analysis and Early Warning Technology of Coal Mine Safety, Anhui Huainan 232001, China

**Abstract: Objective** Traditional conformance detection methods mostly consider the control flow and do not take into account the impact of data on the conformance checking process in process. To address this limitation, a conformance checking method based on the directly following rule model (DFRM) was proposed, extending the traditional directly following model. **Methods** Firstly, the control flow and data flow of the real process model were respectively transformed into business process rules from their own perspectives. Then, the control flow rules were combined with the data flow rules using association rules, and the data rules were processed using satisfiability modulo theory (SMT), ultimately obtaining the representation based on the direct-following rule model. Then, event logs were extracted from event data, and control flow alignment was prioritized before data flow alignment to achieve multi-perspective business process conformance checking. **Results** The proposed method was evaluated using real medical event logs. Compared with other traditional methods, the proposed method has better performance in terms of model fitness and F1 score, and the model

**收稿日期:**2023-07-27 **修回日期:**2023-09-20 **文章编号:**1672-058X(2025)02-0095-10

**基金项目:**国家自然科学基金(61572035);安徽省重点研究与开发计划项目(2022A05020005)。

**作者简介:**左殷恺(1998—),男,江苏连云港人,硕士,从事业务流程多视角异常检测研究。

**通讯作者:**卢可(1995—),男,安徽宿州人,博士研究生,从事过程挖掘、业务流程监控研究。Email:luke0610@outlook.com。

**引用格式:**左殷恺,卢可.业务流程一致性检查方法——基于直接跟随规则模型[J].重庆工商大学学报(自然科学版),2025,42(2):95-104.

ZUO Yinkai, LU Ke. A conformance checking method for business process: Based on directly following rule model[J]. Journal of Chongqing Technology and Business University (Natural Science Edition), 2025, 42(2): 95-104.

has less execution time due to its simplicity. **Conclusion** Combining the data perspective with the directly following model in the form of data association rules provides a new extension direction for the model representation of current business process software. Introducing the data perspective into the business process solves the problem that existing process models cannot express some decision requirements related to data, and also takes into account the possible biases of other perspectives in the conformance checking process.

**Keywords:** conformance checking; association rules; smart healthcare; directly following rule model (DFRM)

## 1 引言

在大数据飞速发展的时代背景下,业务流程管理在许多行业都有着举足轻重的作用。业务流程模型的一致性检测一直是业务流程管理的重要组成部分,由于信息系统设计不足或系统升级,这些系统产生的事件日志可能与现有模型之间存在不一致。而一致性检测技术可以检测到此种不一致,并验证和扩展业务流程模型,从而确保组织内系统的可信度。近年来,有关单个模型与其自身的控制流网和数据流网的研究有很多。传统的流程表现形式有很多,例如 Petri 网、BPMN (Business Process Model Notation) 等,但都局限在控制流。为了顺应多视角数据流程管理的潮流,发展出了 DPN、BPMN-E<sup>2</sup> 等拓展形式,但目前有关直接跟随模型 (DFM) 数据视角的拓展研究却相对较少。

虽然以往关于业务流程一致性检测的研究已有很多,但基本都停留在控制流层面。例如,文献[1]创造性地提出了一种基于重放分析技术的一致性检查方法,不仅能够处理不可观察的活动,而且能够识别过程模型中跳过的活动和事件日志中插入的活动。文献[2]将建模行为和观察到行为关联起来,并假设最小的对齐偏差即为对所观察到的行为的最佳解释。从上述方法中可以得出:传统的业务流程一致性检查方法主要关注流程的活动序列,即控制流层面,而对于流程的其他视角,如数据视角、组织视角等,所产生的不一致却不能被准确地识别和解释。因此在业务流程管理中,将流程的多个视角都纳入考量是必要的。文献[3]首先将流程的控制流进行对齐,为了将对齐技术扩展到其他流程视角(即数据、时间和资源),通过构造额外的 ILP(线性规划问题)进行解决。文献[4]同样在控制流对齐的基础上,通过 CRUD 矩阵对流程模型中数据对象的操作进行限制。规定好过程模型中合法行为后,再将对数据对象的操作与原有的事件日志进行数据流的对齐。上述研究都是优先将控制流视角进行对齐,忽略了数据流的偏差可能导致控制流的不一致,因而文献[5]在之前研究基础上进行了扩展,通过采用成本函数平衡了控制流和其他视角的偏差。方法可以有效避免传统单视角方法产生的误导性结果。

为了处理业务流程中的数据流表示问题,数据规则通常是一种表现形式。文献[6]提出了在控制流关

联规则的基础上进行数据流关联规则的挖掘方法,从数据流角度对事件日志进行异常分析。文献[7]提出了一种基于一阶逻辑片断(FOL)的简单易读的规则语言,并展示了如何直接从 BPMN 模型生成遵从性规则。但其重点在于如何将 BPMN 模型中的控制流视角转换为规则,却并未考虑模型的其他视角,具有一定的局限性。文献[8]关注于特定领域(即危害分析和关键控制点系统(HACCP))的特点,对传统 BPMN 进行拓展,提出了一种名为 BPMN-E<sup>2</sup> 的表示方法,并利用多视角一致性检查方法对业务流程进行分析。文献[9]提出了一种分布式、多视角一致性检测框架,从 BPMN 框架里提取出一致性规则后将之发布在区块链上,以实现可信地一致性检测。对于数据规则的处理,文献[10]提出了一种多视角挖掘技术来发现数据连接,并采用关联规则挖掘的概念来提取事件序列和相关数据对象的行为属性以及上下文特征之间的联系,最后利用事务表对规则集进行简化合并。基于关联规则的挖掘方法也常用于业务流程的异常检测领域。文献[11]提出了一种基于关联规则挖掘的异常检测方法,它促进了对异常现象的解释和对异常现象严重程度的估计,这有助于对恶性异常采取适当的对策,并且避免良性进程执行的可能终止。上述研究的建模语言大多使用 BPMN 和 Petri 网,对于 DFM 的应用研究则相对较少。有关直接跟随模型(DFM)的一致性检查技术在文献[12]中被提出,但其研究却仅停留在控制流层面,并未考虑流程中其他因素对流程分析的影响。有研究提出一种从业务流程事件日志中提取的 DFGs 的泄漏保护方法,以求在差分隐私模型下获取泄漏风险和效用损失之间的平衡。

直接跟随图(DFM)由于其简洁性,受众广泛,因而受到众多商业流程管理软件的青睐,如 Disco、Celonis 等。但以往对其数据视角的拓展研究却相对较少,因此,提出了将 BPMN 转换为直接跟随规则模型(DFRM)的方法。首先将活动的附加数据条件转换为数据规则,由两个活动中的弧负载数据规则;接下来先将控制流进行对齐,再验证事件日志中的数据是否符合活动间的数据规则,最终获得一致性检测结果。

## 2 动机案例

为了更好地展现以数据为中心的业务流程,考虑

使用图 1 中描述的 BPMN 流程模型。此业务流程描述了从控制流和数据流两个层面对疑似糖尿病患者的初步诊断与治疗。如图 1 所示,流程从开始触发到要求护士接收病人,护士是负责接收病人的过程资源。护士通过判断附加在活动上的文本注释,即是否有低血糖或糖尿病酮症酸中毒症状来判断患者的病情是否紧急,并输出患者情况 Pc。如果患者病情紧急则需要护士临时进行预抢救,对生命体征进行维持;如果病人病情不紧急,则需要护士进行预评估,对患者身体状况检查评估并输出评估报告。医生则需要对之前输出的患

者信息 Pc 或者评估报告 Er 进行综合考虑并进行分诊。诊断分为一般会诊以及专家会诊,医生会根据病情要求患者进行预约检查。检查分为一般身体检查和高级检查,最后医生根据检查报告 Ir 和之前的患者情况 Pc,诊断出结果。判断病人的血糖是否在健康阈值内,倘若患者血糖 (Glucose, GLU) 在 4.4 mmol/L 至 10 mmol/L 之内,则需要医生开药物处方并告诉患者未来的治疗计划,在间隔 7 d 后再来医就诊,否则患者需要继续住院治疗观察病情发展,最后需要对相关诊断和检查进行缴费。

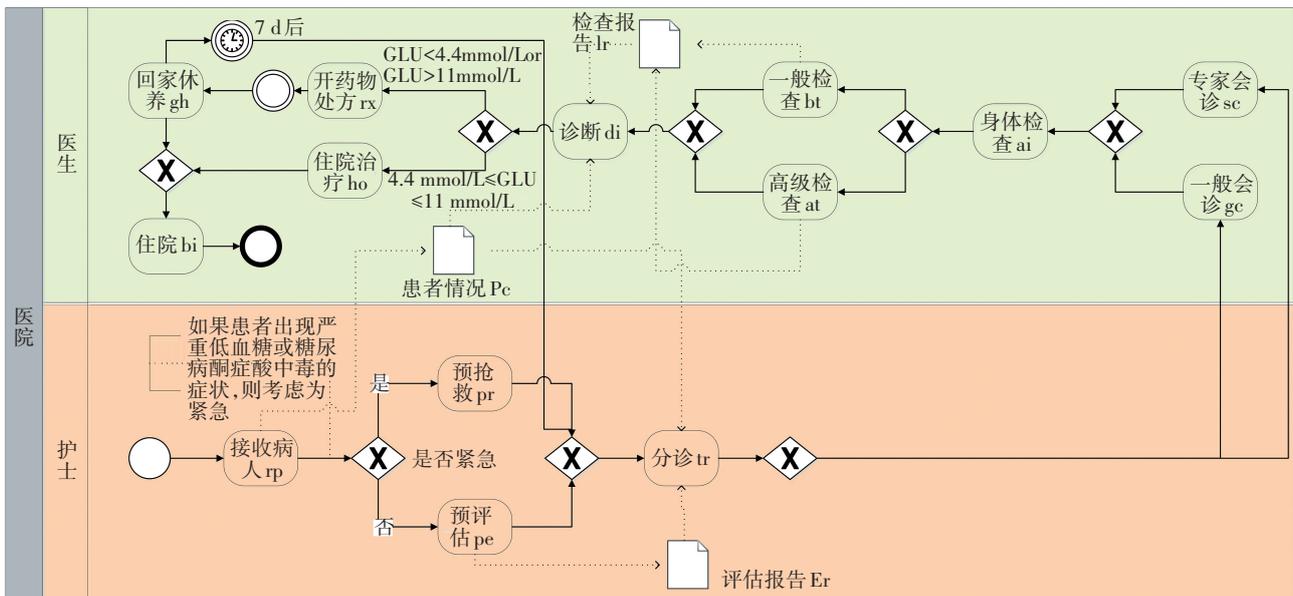


图 1 糖尿病患者就医流程

Fig. 1 Medical process of patients with diabetes

### 3 预备知识

**定义 1** (直接跟随图)<sup>[12]</sup> 事件日志  $L$  的直接跟随图是  $G(L) = \{(D_1, R_1), (D_2, R_2), \dots, (D_n, R_n)\}$ 。其中,  $D = \{(x, y) \mid x \in A \wedge y \in A \wedge (x <_L y)\}$  是一对两个活动,  $R$  是图的边集, 其权值为  $W = \{w \mid w \in \mathbf{R}\}$ , 而  $x <_L y$  成立当且仅当  $\exists t = \langle e_1, e_2, \dots, e_m \rangle$  和  $i \in \{1, 2, \dots, m-1\}$ , 其中  $t \in \text{Trace}(L)$  且  $t_i \cdot a = x, t_{i+1} \cdot a = y$ 。

**定义 2** (业务流程模型的 Petri 网)<sup>[13]</sup> 一个 Petri 网是一个 5 元组  $PN = (P, T, F, A, \iota)$ , 其中

- (1)  $P$  是一个有限的库所集。
- (2)  $T$  是一个有限的变迁集, 且  $P \cap T = \emptyset$ 。
- (3)  $F \subseteq (P \times T) \cup (T \times P)$  是有向弧集。
- (4)  $A$  是一个活动标签且  $v \in A$  表示一个不可见标签。
- (5)  $v: T \rightarrow A$  是一个标记函数, 它将一个变迁映射到一个标签。

**定义 3** (对齐)<sup>[14]</sup> 给定一个日志迹  $\sigma = \langle \sigma_1, \dots, \sigma_m \rangle \in L$  和一个过程模型  $N = \langle P, T, F, m_0, m_f, \Sigma, \lambda \rangle$ ,  $\sigma$  与  $N$  的对齐是一个有限序列参数  $\varphi = \langle (\sigma'_1, u'_1), \dots,$

$(\sigma'_p, u'_p) \rangle$  这样的移动。

(1) 移动分为同步移动  $(a, t) \in \Sigma \times T$ , 其中  $a = \lambda(t)$ ; 日志移动  $(a, \gg)$  (其中“ $\gg$ ”是一个特殊的“跳过”符号); 或者模型移动  $(\gg, t)$ 。

(2) 从左投影  $\varphi$  的  $\langle \sigma'_1, \dots, \sigma'_p \rangle$  中删除“ $\gg$ ”便会产生  $\sigma$ 。

(3) 从右投影  $\varphi$  的  $\langle u'_1, \dots, u'_p \rangle$  中删除“ $\gg$ ”便会产生  $N$  的完整运行  $u$ 。

**定义 4** (经典对齐成本函数, 最优对齐)<sup>[14]</sup> 对于模型  $N$  和日志迹  $\sigma$  之间的每个对齐  $\varphi$ , 经典对齐成本函数  $C$  将成本 0 分配给同步移动, 将成本 1 分配给日志移动或模型移动。对齐的成本是其移动成本的总和, 如果模型  $N$  和日志迹  $\sigma$  之间的对齐使成本最小化, 则它是最佳的。

### 4 模型设计与构建

主要介绍了基于数据规则的直接跟随模型一致性分析方法, 并提出了流程到 DFRM 的转换算法和多视角业务流程的一致性检测方法。旨在促进流程的快速

对齐,为减少业务流程的一致性检测分析时间做出贡献。如图 2 所示,主要有 3 个部分。其中红框内表示对原始事件数据的提取阶段,对信息系统中的事件数据进行提取;绿框内为对业务流程(BPMN)进行转换,将

较为复杂的高级建模语言转换为 DFM,将相关数据条件按照算法 1 转换为数据规则;蓝框内表示对 DFRM 进行多视角的一致性检测,先后对控制流和数据流进行对齐,最后得出一致性结果。

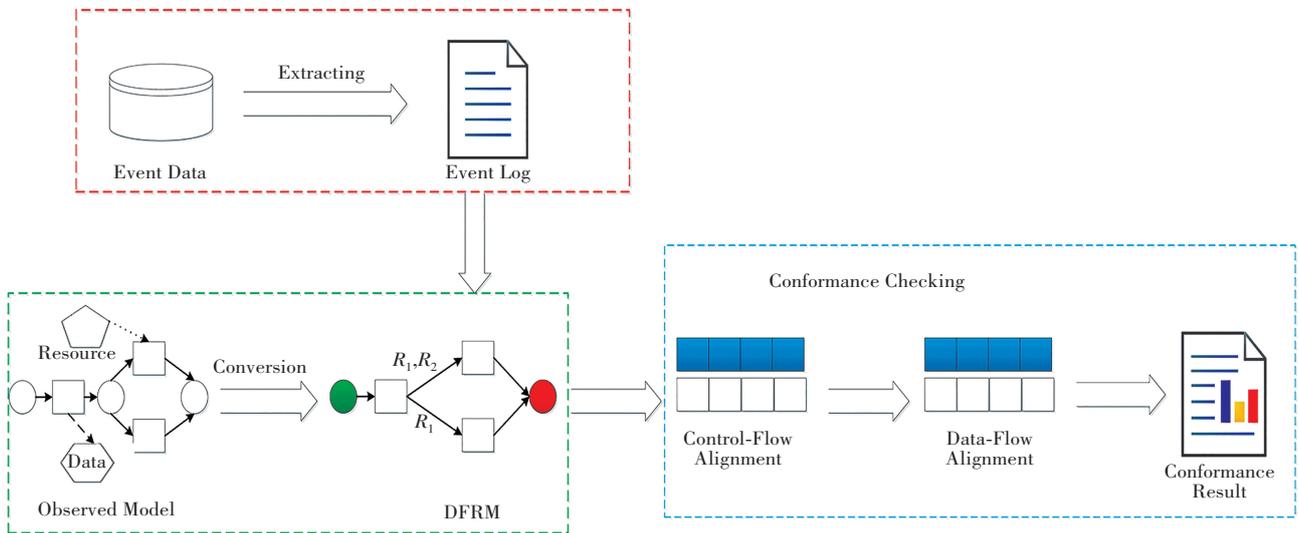


图 2 一致性检测流程图

Fig. 2 Flowchart of conformance checking

4.1 业务流程到 DFRM 的转换

4.1.1 转换阶段

定义 5 (过程模型) 一个过程模型  $M = (N_p, DN, C, F, re, \rho, A, tr, G)$  是一个九元组, 包含:

(1) 有限的非空控制流结点  $N_p$ 、有限的数据节点  $DN$ 、有限的非空控制流边集  $C$ 、有限的数据关联集  $F$ 、有限的资源集合  $re$  以及活动集合  $A$ ,  $G$  代表变迁的守卫, 其中  $N_p$  包含普通变迁结点  $N_T$  和用户执行结点  $N_{UT}$ 。

(2) 函数  $\rho: A \rightarrow re$  为每个用户执行结点活动分配负责其执行的资源。

(3) 控制流  $C \subseteq N_p \times N_p$  连接  $N_p$  的元素, 数据节点  $DN$  由不相交集数据对象  $DO$  和对活动进行时间限制的数据结点  $tr$  组成;

(4)  $F \subseteq (DN \times A) \cup (A \times DN)$  是连接数据节点和活动的数据关联集, 其中  $F$  包含  $T_{in}$ , 用于将数据元素链接

到活动或事件的输入关联关系,  $T_{out}$  用于将数据元素链接到活动或事件的输出关联关系。

定义 6 (DFRM) 直接跟随规则模型可以由一个三元组  $DFRM = (N_d, E, R)$  表示。其中  $N_d$  为有限的控制流结点, 包含开始结点集  $N_{start}$ 、结束结点集  $N_{end}$  及中间结点集  $E_{int}$ , 即  $N_d = \{N_{start} \cup N_{int} \cup N_{end}\}$ ;  $E \subseteq N_d \times N_d$  为模型的边集;  $R$  为模型的规则集。

如图 3 所示, 给出了流程的一致性规则生成过程。首先对 BPMN 模型进行控制流与数据约束的分离, 控制流直接转化为 DFM, 而 BPMN 的数据约束通过算法 1 将之转换为 BPMN 约束规则集; 然后将 DFM 的控制流规则集与 BPMN 的约束规则集通过算法 2 进行合并, 最后生成流程的一致性规则。下面将详细介绍算法 1 和算法 2 的主要思想, 而对于数据规则的处理, 文献[15]提出了 SMT 技术, 对一阶逻辑公式进行解决并进行一致性检测, 将在 4.2.2 小节做简单介绍。

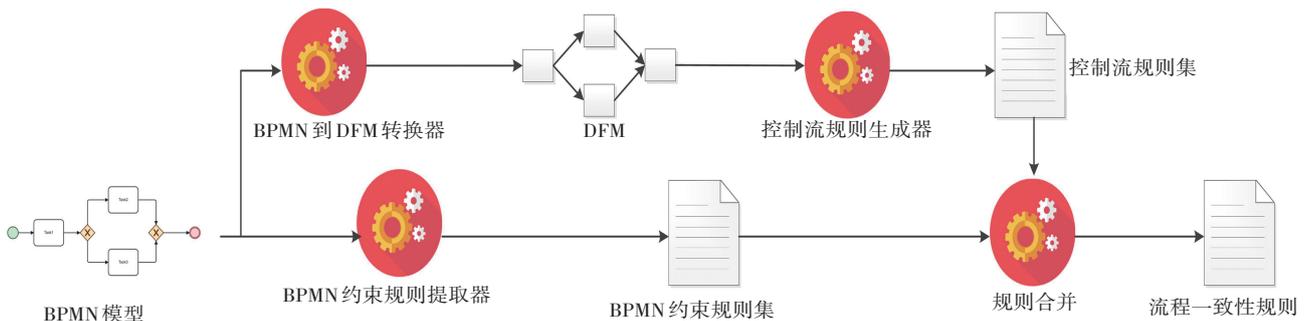


图 3 流程一致性规则生成过程

Fig. 3 Process of generating conformance rules

## 算法1: BPMN 约束规则提取算法

输入: 过程模型 BPMN

输出: BPMN 约束规则集

```

(1) begin: LaneRules ← {} GETLANERULES()
    TimedRules ← {} GETTIMEDRULE();
(2)   ResourceRules ← {} GETRESOURCERULES();
(3)   GuardRules ← {} GUARDRULES()   end;
(4) procedure GETLANERULES()
(5)   for each re ∈ BPMN. lanes do
(6)     rule ← ();
(7)     rule. data ← (User = re. name);
(8)     for each a ∈ re. A do
(9)       LaneRules[a. name] ← rule   end procedure;
(10) procedure GETTIMEDRULES()
(11)  for each a ∈ BPMN. A do
(12)    Timers ← GETPRECEDINGTIMERS(a);
(13)    for each t ∈ Timers do
(14)      pTasks ← GETPRECEDINGACTIVITIES(t);
(15)      nTasks ← GETSUCCEEDACTIVITIES(t);
(16)      rule ← ();
(17)      rule. data ← (nTasks. start - pTasks. end ≥ Timers.
        value);
(18)      TimedRules[a. name]. add(rule)   end procedure;
(19) procedure GETRESOURCERULES()
(20)  for each do ∈ BPMN. DO do
(21)    Tasks ← GETRELATEDACTIVITIES(do);
(22)    for each a ∈ Tasks do
(23)      rule ← ();
(24)      if F. Tin = true then
(25) rule. data ← ("read"+do) or rule. data ← ("create"+do);
(26)      else rule. data ← ("update"+do);
(27)      ResourceRules[a. name]. add(rule)   end procedure;
(28) procedure GETGUARDRULES()
(29)  for each a ∈ BPMN. A do
(30)    Guard ← GETPRECEDINGGUARD(a);
(31)    for each guard ∈ Guard do
(32)      rule ← ();
(33)      rule. data ← (G = guard. value)
(34)      GuardRules[a. name]. add(rule) end procedure;
(35) return LaneRules, TimedRules, ResourceRules, GuardRules

```

## 算法2: 规则合并算法

输入: 控制流规则集 (CFR),

LaneRules, ResourceRules,

TimedRules, GuardRules

输出: 流程规则集 R

```

(1) begin processRules ← {};
(2)  for each cRule ∈ CFR do
(3)    event ← cRule. key;
(4)    rule ← cRule. value;
(5)    if TimedRules. haskey(event) then
(6)  for each tempRule ∈ TimedRules. get(event) do
(7)    taskName ← tempRule. task;
(8)    oldRule ← rule. getTaskRule(taskName);
(9)    rule ← oldRule. add(tempRule);
(10)   if LaneRules. haskey(event) then
(11)     rule. add(LaneRules. get(event));
(12)   if ResourceRules. haskey(event) then
(13)     rule. add(ResourceRules. get(event));
(14)   if GuardRules. haskey(event) then
(15)     rule. add(GuardRules. get(event));
processRules[event] ← rule   end;

```

算法1的主要思想是通过将 BPMN 中各种不同的数据条件通过对应的转换函数,转换为数据约束规则。业务流程中主要分为4种数据条件:资源执行者、数据对象、守卫和定时数据对象。1—3行首先将4种规则集设置为空集并定义转换数据规则转换函数;4—9行为获取资源执行者的数据规则转换函数,其中5—7行为获取 BPMN 中的 lanes 集合,即定义5中的资源执行者 re 的全集,8—9行将每个有资源执行的活动赋值上不同的资源执行者;10—18行为获取时间限制的规则函数,其中第12行是为了获取活动开始前的定时数据对象,第13—15行分别获取改定时数据对象的前后活动,第17行规则是要判断后缀活动的开始时间减去前缀活动的结束时间是否大于定时数据对象的值,最后18行将规则添加到 TimedRules 里;19—27行是获取数据对象规则的函数,其中,20—21行是为了获取与数据对象有关的活动,通过获取相关活动函数 GETRELATEDACTIVITIES() 获取,该函数通过定义5中的数据关联规则集 F 来寻找相关活动;22—26行通过判断 F 的输入/输出关联关系分别产生读写数据对象规则;第27行将规则添加到 ResourceRules 里;28—34行是对守卫进行转换的函数,其中29—33行为确定活动前的守卫并将守卫值赋给规则,最后将约束规则添加到函数 GuardRules 里。

算法2的主要思想是将 DFM 产生的控制流规则与

算法 1 转换后的 BPMN 约束规则集进行合并,生成最终的流程一致性规则集。将算法 1 中生成的 4 种规则集 LaneRules、ResourceRules、TimedRules、GuardRules 和 DFM 的控制流规则集(CFR)作为输入。首先,2—4 行将 CFR 中的活动和活动间的关系分别赋值于 event 和 rule,然后判断 TimedRules 中是否有相关的活动,然后 5—9 行将 TimedRules 中的活动与 CFR 中的活动进行对应并将数据约束规则重新赋值给新的规则集,10—15 行分别对另外 3 种规则集与之前的控制流活动进行合并,最后 16 行将之前的规则合并生成流程规则集  $R$ 。规则的形式是利用关联规则将 DFM 的控制流规则与算法 1 提取的数据约束进行组合(即算法 2)。如图 1 所示,执行接受病人  $rp$  活动后判断病人病情是否严重,倘若严重,则活动预抢救  $pr$  的守卫为真,该业务流程片段的规则为  $IF \sigma(i). a=rp \wedge \sigma(j). a=pr \text{ THEN } \sigma(j). guard=True.$

#### 4.1.2 SMT 解决一致性规则

满足性模理论(Satisfiability Modulo Theories),简称 SMT 技术,可以用于解决检查一个逻辑公式是否满足。文献[16]介绍了 SMT 技术的基础知识和相关挑战,其重要的应用领域便是形式化验证<sup>[17]</sup>。因其具有逻辑性质,因而一致性规则的验证问题可以看作是一个 SMT 问题。因此,一致性规则可以被看作是一个 SMT 求解器,它使用业务流程元素条件的初始数据条件集实例化。在一致性检查过程中,可以根据事件日志数据添加一组新的数据条件并解决 SMT 问题来验证这些规则。在满足的情况下,规则也被满足,否则规则不被验证。

例如图 1 中,活动住院治疗  $ho$  的初始数据条件为  $GLU < 4.4 \text{ mmol/L}$  或  $GLU > 10 \text{ mmol/L}$ ,这个为需要检查的一致性规则。倘若流程运行到此活动时,  $GLU$  为  $11 \text{ mmol/L}$ ,此为事件日志提供的实际值。最后,需要检查规则。在这种情况下,由于提供的  $GLU$  大于  $10$ ,因此满足一致性规则。

#### 4.2 基于 DFRM 的一致性分析

**定义 7** (合法移动集)<sup>[18]</sup> 一个合法移动由一对  $(s_L, s_M) \in \mathcal{E}^{\gg} \times S_M^{\gg}$  表示,其中  $s_L = (a_L, D, c, i)$ ,  $s_M = ((\lambda, (p, a_M, p')), w)$ ,其中  $p = (P, \varphi)$  和  $p' = (P', \varphi')$ ,使得:

- (1)  $(s_L, s_M)$  是一个日志移动,如果  $s_L \in \mathcal{E}$  且  $s_M = \gg$ 。
- (2)  $(s_L, s_M)$  是具有满足守卫的模型移动,如果  $s_M \in S_M$  和  $s_L = \gg$ ,并且数据守卫  $G(s_M)$  在  $\varphi$  和  $\varphi'$  上评估为真。
- (3)  $(s_L, s_M)$  是具有违反守卫的模型移动,如果  $s_M \in S_M$  和  $s_L = \gg$ ,并且数据守卫  $G(s_M)$  在  $\varphi$  和  $\varphi'$  上评估为假。
- (4)  $(s_L, s_M)$  是满足守卫的两个移动,如果  $s_L \in \mathcal{E}$ ,  $s_M \in S_M$  和  $a_L = a_M$  且  $G(s_M)$  在  $\varphi$  和  $\varphi'$  上评估为真。
- (5)  $(s_L, s_M)$  是违反守卫的两个移动,如果  $s_L \in \mathcal{E}$ ,

$s_M \in S_M$  和  $a_L = a_M$  且  $G(s_M)$  在  $\varphi$  和  $\varphi'$  上评估为假。

**定义 8** (代价函数) 设  $(s_L, s_M)$  是日志迹和模型迹之间的移动,定义了一个衰减函数  $\mu(i, j, l, m)$  对流程中活动的移动进行惩罚。其中  $i$  为日志迹中活动所在位置数,  $j$  为模型迹中活动所在位置数,  $l$  为日志迹整体的活动数,  $m$  为模型迹的整体活动数。设置了函数

$$f(\tau_l, \tau_m) = \frac{\sum_{r \in sr} C_l}{\sum_{r \in R} C_m}, \text{ 其中 } \tau_l, \tau_m \text{ 分别为日志和模型的迹,}$$

$r$  为单条规则,  $sr$  为满足的规则集,  $R$  为流程整体规则集,  $C_l, C_m$  分别为日志和模型中的流关系。成本函数  $\kappa(s_L, s_M)$  被定义为

$$\kappa(s_L, s_M) = \begin{cases} 1 + \mu(i, j, l, m) & \text{满足规则的日志/模型移动} \\ 2 + \mu(i, j, l, m) - f(\tau_l, \tau_m) & \text{违反规则的模型移动} \\ 1 - f(\tau_l, \tau_m) & \text{违反规则的同步移动} \\ 0 & \text{同步移动} \end{cases}$$

$$\text{其中, } \mu(i, j, l, m) = \begin{cases} \frac{i-1}{l}, & \text{日志移动} \\ \frac{j-1}{m}, & \text{模型移动} \end{cases}.$$

在文献[19]中,折扣函数对于流程中前面的活动惩罚力度比后面的惩罚力度更大。例如图 1 中的实例,对于前面的活动判断如病情是否严重,活动  $pr$  和活动  $pe$  对于流程的整体影响更大,而活动  $ho$  和  $pf$  对于整体流程影响相对较小。因此,对于经典的代价函数做了改动,旨在不影响对齐速度的同时更加贴切现实流程。

结合表 1 的示例日志与第 2 节所示的流程模型对算法 1 中提到的几种流程规则进行解释。在确定控制流已经对齐的前提下,对数据层面进行分析。示例日志中,事件开药物处方  $rx$  的执行者为护士,而模型中执行者为医生,这显然产生了冲突,违反了算法 1 中的 LaneRules。对于模型中的两个守卫,需要判断流程执行的活动与守卫之间是否冲突。正常执行守卫 1 时,判断为 True 时执行  $pr$ ,判断为 False 时执行  $pe$ 。从日志中可以判断守卫 1 满足规则。而对于守卫 2 则是需要判断  $GLU$  的值是否在合理区间。由于在活动诊断  $di$  之后发生了开药物处方  $rx$ ,因而需要  $GLU < 4.4 \text{ mmol/L}$  或者  $GLU > 11 \text{ mmol/L}$ 。而日志中守卫 2 为  $5.6 \text{ mmol/L}$ ,明显违背了 GuardRules 数据规则。对于定时数据规则,要判断活动  $gh$  和活动  $tr$  之间的间隔是否满足大于  $7 \text{ d}$  这个规则。对于数据对象,传统的采用 Crud 矩阵来关联数据对象和活动之间的关系<sup>[20]</sup>,本文将之简化为  $Cwr$ 。流程中有 3 个数据对象,包括患者情况  $Pc$ 、评估报告  $Er$  和检查报告  $Ir$ 。日志中  $Er$  和  $Ir$  符合流程规则,而对于患者情况  $Pc$ ,活动诊断  $tr$  并没有对它的更新权,因此产生错误,违反了 ResourceRules。

表 1 示例日志  
Table 1 Example log

案 例	活 动	时 间	道 (lanes)	守 卫 1	守 卫 2	患 者 情 况 (Pc)	评 估 报 告 (Er)	检 查 报 告 (Ir)
1	rp	2023.1.1 9:00	nurse			Cw		
1	pe	2023.1.1 9:10	nurse	False			Cw	
1	tr	2023.1.1 9:30				wr	r	
1	gc	2023.1.1 9:50	doctor					
1	ai	2023.1.1 10:00						
1	bt	2023.1.1 11:00						Cw
1	di	2023.1.1 11:20	doctor		5.6	r		r
1	rx	2023.1.1 11:30	nurse					
1	gh	2023.1.1 11:40						
1	bi	2023.1.1 12:00						

### 5 仿真实验与结果分析

为评估上述方法的有效性,主要通过 3 个角度进行分析:使用 DFRM 进行业务流程一致性检查,验证其相对于其他传统方法在执行时间方面是否有所提升;新方法在模型质量度量指标上与其他方法的比较;随着事件日志大小的增加,新方法与其他方法在性能上的比较。

#### 5.1 实验设计

为了实现上述 3 个实验目标,实验主要从 2 个阶段进行解释。实验是在配备 3.4 GHz 英特尔酷睿 i7 处理器和 16 GB 内存的机器上进行的。输入的数据集是一个合成的事件日志,包含频繁和不频繁的迹。图 4 显示了应用于实验的医疗事件日志,其中包含 1 000 条迹、16 736 个事件、14 个事件类和 9 个属性类。



图 4 事件日志  
Fig. 4 Event log

实验的第一阶段将针对第 1 个和第 2 个实验目标进行设计解释。为了评估方法一致性检测结果的准确性和运行效率,应用 4 种不同模型一致性检测方法,通过添加不同数量比例的噪声评价其对一致性检测结果的影响。分别计算精确度(Precision)、查全率(Recall)和 F1-score,评价一致性检测的准确性,并测量一致性检测的运行时间。

实验将本文的 DFRM 方法与传统的 DFM (Directly Follow Miner)、HM (Heuristic Miner) 和 A\* with ILP 方法进行对比。其中 DFM 通过 Prom<sup>[21]</sup> 框架中的 DFVM (Directly Follow Visual Miner) 插件实现。如图 5 所示,方法只能实现传统控制流视角的一致性检测,但没有考虑实际业务流程执行过程中约束条件和数据属性对业务流程一致性检测的影响。

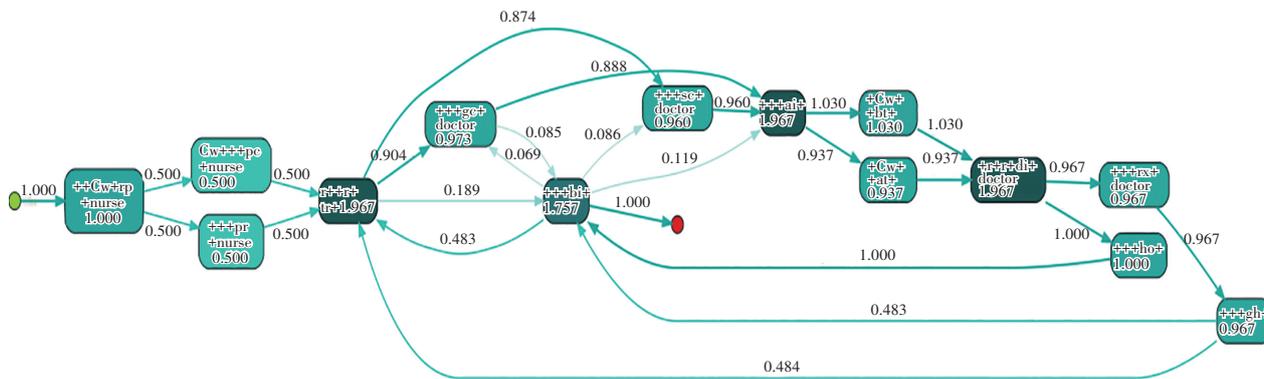


图 5 直接跟随挖掘器

Fig. 5 Directly follow visual miner

考虑 DFM 方法的局限性,为了保证实验的科学性,对于 HM 和 A \* with ILP 方法将都考虑数据视角。HM 方法通过 Prom 框架中的 iDHM(interactive Data-aware Heuristic Miner)<sup>[22]</sup> 插件实现。A \* with ILP 方法使用 Multi-perspective Process Explorer<sup>[23]</sup> 插件实现,如图 6 所示,并且该插件可以实现对控制流和数据流的一致性检测,并反映控制流与数据流各自的不一致性比率。

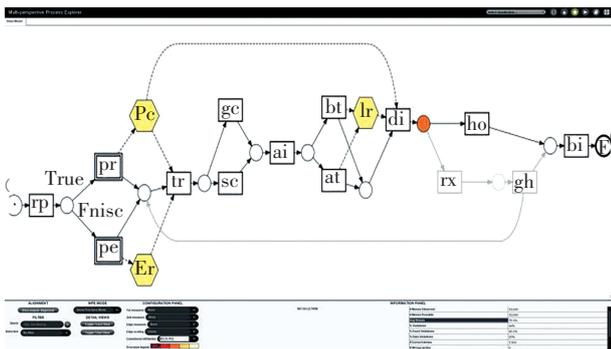


图 6 多视角流程浏览器

Fig. 6 Multi-perspective process browser

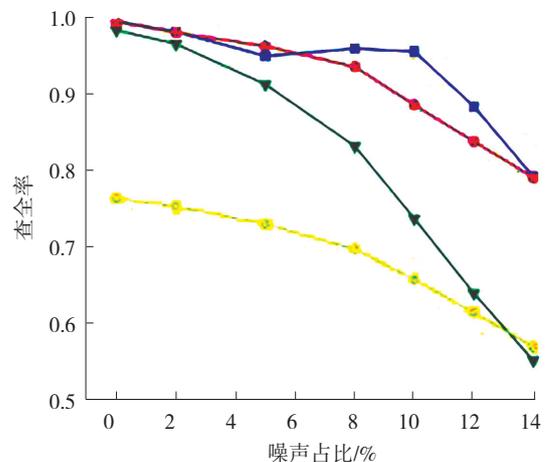
实验的第二阶段,逐步增加日志迹的数量,探究新方法对于较大日志的表现,并与传统方法进行比较。以 1 000 条迹为步长,输入 7 个事件日志(1 000—7 000),探究 4 种方法在执行时间上的表现。

### 5.2 实验结果分析

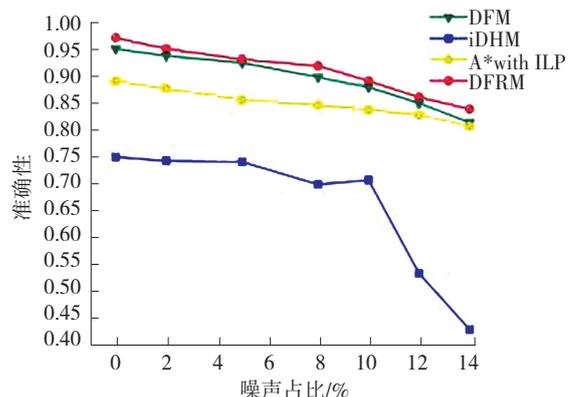
实验的第一阶段通过在日志中注入不同数量级的噪声,评估不同事件日志下,4 种方法对一致性检测结果的影响。图 7(a)显示了噪声对 4 种方法查全率的影响。噪声包含控制流和数据流的偏差,其中控制流偏差包含活动的缺失、附加和错位的噪声,而数据噪声包含资源类型的错位和活动与数据执行不正确匹配的噪声。随着噪声含量的增加,4 种方法一致性检测结果的召回率都呈下降趋势。提出的 DFRM 方法优于 DFM、A \* with ILP 和 iDHM,且在噪声低于 5% 时具有较好的表现,随着噪声的增加表现略低于 iDHM 方法。相对于其他 3 种方法,DFRM 比 DFM 多考虑数据视角,因此在拟合度上会有较高的表现。A \* with ILP 方法的执行表现是 4 种方法中较差的,如图 8 所示,展示了 Multi-perspective Process Explorer 在一致性检查中控制流和数据流各自的违反比例以及总的违反百分比。可以看出随着噪声的增加,总的违反占比受控制流违反影响较大。而启发式挖掘(HM)方法本身对于噪声的鲁棒性就较高,在考虑到数据视角后执行表现有略微下降。图 7(b)描述了随着活动和数据噪声的增加,4 种方法

的准确性逐渐下降。随着噪声的增加,DFM 方法粗暴地删除不符合业务流程模型的日志,保留了模型中的大部分行为。因此,精度保持在 0.9 左右,实际上存在过拟合问题。与 iDHM 相比,随着噪声的增加,新方法将活动视图与数据规则相结合,准确检测多视角的一致性问题,且准确率下降缓慢,最小准确率为 0.838 5。现实生活中可能两个指标会存在冲突的情况,因此图 7(c)引用了 F1 分数指标进行衡量,它是 Fitness 与 Precision 的权衡度量指标,可以表现一个模型的综合性能,而新方法具有明显的优势。图 7(d)显示了随着噪声含量的增加,4 种方法执行时间的总体增加趋势。本文的方法相对于 iDHM 具有较少的执行视角,且对于其他两种方法运行时间的增幅也不是很大。

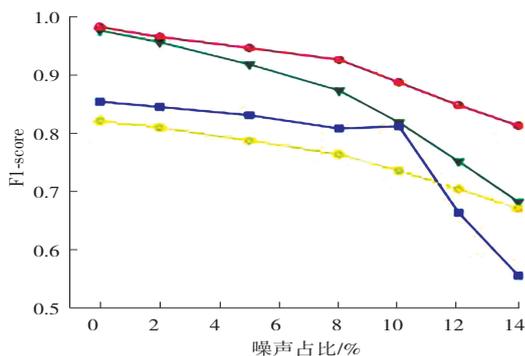
图 9 为事件日志大小对执行时间的影响。从图 9 可以看出:随着事件日志的大小逐渐增加,没有考虑数据视角的 DFM 执行时间最少,而 iDHM 的执行时间最长。由于新方法使用关联规则与控制流视角结合,数据规则可以较好地转换为声明式规则,可以较快地执行。因此新方法在较大的事件日志情况下,仍然具有较高的执行效率。



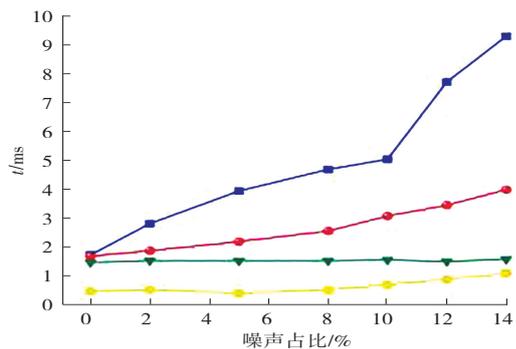
(a) 对查全率的影响



(b) 对准确性的影响



(c) 对 F1-score 的影响



(d) 总体趋势

图 7 噪声对一致性结果影响

Fig. 7 Impact of noise on conformance results

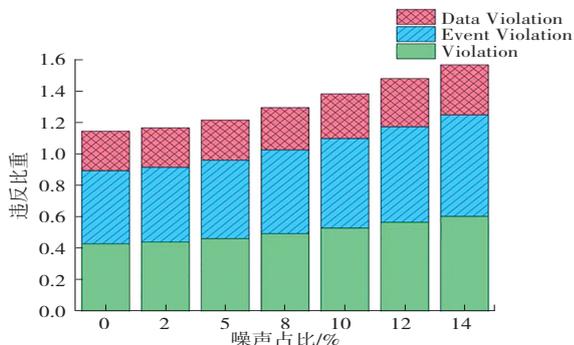


图 8 多视角流程浏览器一致性违反比重

Fig. 8 Weight of consistency violation of multi-perspective process browser

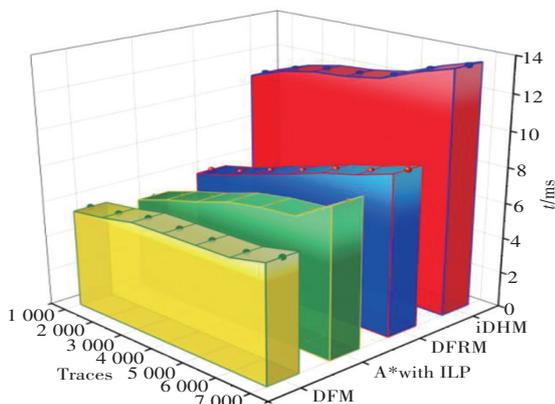


图 9 事件日志大小对执行时间的影响

Fig. 9 Impact of event log size on execution time

综上所述,提出的基于 DFRM 的一致性检测方法,相对于其他 3 种方法具有较高的拟合度和精确度,且随着事件日志的增大仍然具有较短的运行时间,体现了方法在具有较大事件日志时,仍具有较高的执行效率。

### 6 结论

主要贡献在于扩展了传统的直接跟随图,将数据视角以数据关联规则的形式与直接跟随模型相结合。这对于目前的业务流程商业软件的模型展现形式提供了新的扩展方向。将数据视角引入到业务流程中,解决了现有流程模型无法表达与数据相关的一些决策需求的问题,未能发现一致性检测过程中其他视角的可能偏差。首先,将现实流程模型的控制流和数据流分别转换为各自视角的数据规则;然后,将控制流规则与数据流规则相结合,最终实现基于 DFRM 的表现形式;最后,根据不同视角的偏离对业务流程一致性检测结果的影响程度,应用偏离成本函数,检测出数据流和控制流一致性检测过程中的偏差。

使用合成医疗事件日志评估了本文的方法,以较高的效率实现业务流程的一致性检测,保持了较高的拟合度、精确度和较少的执行时间。由于直接跟随图本身存在控制流上的缺陷<sup>[24]</sup>,且关联数据规则的形式较少。在未来工作中,可以寻求执行效率高且具有较好表现形式的模型,在数据规则发现上采用自动流程规则发现可能弥补本文方法的不足。

### 参考文献(References):

[1] ADRIANSYAH A, VAN DONGEN B F, VAN DER AALST W M P. Conformance checking using cost-based fitness analysis [C]// Proceedings of the IEEE 15th international enterprise distributed object computing conference. Piscataway: IEEE Press, 2011: 55-64.

[2] ADRIANSYAH A. Aligning observed and modeled behavior [J]. Technische Universiteit Eindhoven, 2014: 33-117.

[3] DE LEONI M, VAN DER AALST W M P. Aligning event logs and process models for multi-perspective conformance checking: an approach based on integer linear programming [C]// Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013: 113-129.

[4] 胡伟. 多视角事件日志异常分析与一致性检测 [D]. 淮南: 安徽理工大学, 2022.

HU Wei. Multi-view event log anomaly analysis and

- consistency detection [D]. Huainan: Anhui University of Technology, 2022.
- [5] MANNHARDT F, DE LEONI M, REIJERS H A, et al. Balanced multi-perspective checking of process conformance[J]. *Computing*, 2016, 98(4): 407–437.
- [6] 胡伟. 基于多视角关联规则的事件日志异常分析及过滤方法[J]. *现代信息科技*, 2023, 7(2): 69–72, 78.  
HU Wei. Event log anomaly analysis and filtering method based on multi-view association rules[J]. *Modern Information Technology*, 2023, 7(2): 69–72, 78.
- [7] AL-ALI H, DAMIANI E, AL-QUTAYRI M, et al. Translating BPMN to business rules[C]//*Data-Driven Process Discovery and Analysis: 6th IFIP WG 2.6 International Symposium, SIMPDA 2016*, Graz Austria: Springer International Publishing, 2018: 22–36.
- [8] CALHENO R, CARVALHO P, RITO LIMA S, et al. Multi-perspective conformance checking applied to bpmn-e2 [C]//*Trends and Applications in Information Systems and Technologies*. Cham: Springer International Publishing, 2021: 394–404.
- [9] SULIMAN A T, KADADHA M, MIZOUNI R, et al. Blockcheck: a consortium blockchain-based conformance checking framework for business processes [J]. *Internet of Things*, 2023, 21: 100652.
- [10] BAYOMIE D, REVOREDO K, MENDLING J. Multi-perspective process analysis: mining the association between control flow and data objects[C]//*International Conference on Advanced Information Systems Engineering*. Cham: Springer International Publishing. 2022: 72–89.
- [11] BÖHMER K, RINDERLE-MA S. Mining association rules for anomaly detection in dynamic process runtime behavior and explaining the root cause to users [J]. *Information Systems*, 2020, 90: 101438.
- [12] LEEMANS S J J, POPPE E, WYNN M T. Directly follows-based process mining: exploration & a case study [C]//*Proceedings of the International Conference on Process Mining (ICPM)*. Piscataway: IEEE Press, 2019: 25–32.
- [13] CHENG L, LIU C, ZENG Q. Optimal alignments between large event logs and process models over distributed systems: An approach based on Petri nets [J]. *Information Sciences*, 2023, 619: 406–420.
- [14] PEGORARO M, UYSAL M S, VAN DER AALST W M P. Conformance checking over uncertain event data[J]. *Information Systems*, 2021, 102: 101810.
- [15] ANSÓTEGUI C, BOFILL M, PALAHI M, et al. Satisfiability modulo theories: An efficient approach for the resource-constrained project scheduling problem[C]//*Proceedings of the 9th Symposium on Abstraction, Reformulation and Approximation (SARA 2011)*. 2011: 2–9.
- [16] ALT L, ASADI S, CHOCKLER H, et al. HiFrog: SMT-based function summarization for software verification[C]//*Tools and Algorithms for the Construction and Analysis of Systems: 23rd International Conference*. Sweden: Springer Berlin Heidelberg, 2017: 207–213.
- [17] FELLI P, GIANOLA A, MONTALI M, et al. Cocomot: conformance checking of multi-perspective processes via smt[C]//*Lecture Notes in Computer Science*. Cham: Springer International Publishing, 2021: 217–234.
- [18] ZHANG S, GENGA L, DEKKER L, et al. Fuzzy multi-perspective conformance checking for business processes [J]. *Applied Soft Computing*, 2022, 130: 109710.
- [19] BOLTEHAGEN M, CHATAIN T, CARMONA J. A discounted cost function for fast alignments of business processes[C]//*Lecture Notes in Computer Science*. Cham: Springer International Publishing, 2021: 252–269.
- [20] ALIZADEH M, LU X, FAHLAND D, et al. Linking data and process perspectives for conformance analysis [J]. *Computers & Security*, 2018, 73: 172–193.
- [21] LEEMANS S, POPPE E, WYNN M. Directly follows-based process mining: A tool [C]//*Proceedings of the ICPM Demo Track 2019*. Sun SITE Central Europe, 2019: 9–12.
- [22] MANNHARDT F, DE LEONI M, REIJERS HA. Heuristic mining revamped: An interactive, data-aware, and conformance-aware miner [C]//*Bpm (demos)*. 2017.
- [23] MANNHARDT F, DE LEONI M, REIJERS H A. The multi-perspective process explorer [J]. *BPM (Demos)*, 2015, 1418: 130–134.
- [24] VAN DER AALST W M P. A practitioner's guide to process mining: limitations of the directly-follows graph [J]. *Procedia Computer Science*, 2019, 164: 321–328.

责任编辑:田 静