计算机科学
COMPUTER SCIENCE

一种基于带标签时间约束Petri网扩展可达图的数据流通合规性检测

刘振宇, 董慧, 李华, 王璐

引用本文

刘振宇, 董慧, 李华, 王璐. 一种基于带标签时间约束Petri网扩展可达图的数据流通合规性检测[J]. 计算机科学, 2023, 50(11A): 221000118-12.

LIU Zhenyu, DONG Hui, LI Hua, WANG Lu. Compliance Check Method for Data Flow Process Based on Extended Reachability Graph withLabeled Timing Constraint Petri Net [J]. Computer Science, 2023, 50(11A): 221000118-12.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.221000118

一种基于带标签时间约束 Petri 网扩展可达图的数据流通合规性检测

刘振宇　董　慧　李　华　王　璐
内蒙古大学计算机学院　呼和浩特 010021
(cslihua@imu.edu.cn)

摘　要　随着社会制度的不断完善和法律法规的不断健全,企业的经营管理流程面临越来越多的合规性检测要求。利用带标签的时间约束 Petri 网(LTCPN)模型描述数据流通过程中所遵循的法律法规及行业规则。为了支持更多维度的规则表达,首先需要基于 LTCPN 可达图构造扩展可达图 GNR,然后根据含时间戳的事件日志自动生成实际数据流通模型 GNP。通过检测 GNP|=GNR 是否成立来判断基于含时间戳的事件日志的数据流通过程是否符合 LTCPN 描述的规则规范。针对语义信息不明的流程模型合规性检测问题,利用图的点与边连接结构是否相同来检测事件语义无关的功能性属性是否合规。对于语义信息明确的流程模型,可以通过节点或边的语义信息有效减少检测过程中探索的状态空间数量,同时可以进一步丰富合规性检测的非功能性属性检测。通过实验验证了该方法在进行合规性检测方面的可行性。

关键词:Petri 网;时间标签;可达图;图同构;合规性检测
中图法分类号　TP311

Compliance Check Method for Data Flow Process Based on Extended Reachability Graph with Labeled Timing Constraint Petri Net

LIU Zhenyu,DONG Hui,LI Hua and WANG Lu
College of Computer Science,Inner Mongolia University,Hohhot 010021,China

Abstract　With the continuous improvement of social system and laws and regulations,the business management process of enterprise is facing more and more requirements of compliance check. The labeled timing constraint Petri net(LTCPN) model is used to describe the laws,regulations and industry rules followed in the process of data flow. In order to support the rule expression of more dimensions,firstly,it is necessary to construct extended reachability graph GNR based on LTCPN reachability graph,and then automatically generate actual data flow model GNP according to the timestamp event log trace. By examining whether GNP|=GNR to determine that the data flow process based on the event log of timestep is conform to the rule specification described by LTCPN. For the problem of process model compliance check with unknown semantic information,same connection structure of node and edge can be used to detect the functional attribute compliance of semantically independent event. In terms of process models for explicit semantic information,the semantic information of nodes or edges can effectively reduce the number of state spaces explored in checking process,and further enrich the non-functional attribute check of compliance check. The feasibility of method in compliance check is verified by experiments.

Keywords　Petri net,Time-labeled,Reachability graph,Graph isomorphism,Compliance check

1　引言

企业在经营管理过程中面临着越来越多的法律法规需要遵循,如《Sarbanes-Oxley Act》(SOX,萨班斯-奥克斯利法案)[1]和《Basel III》(巴塞尔协议 III)[2]。这些要求所有的组织要审计自己的业务过程,并确保这些业务符合法律和法规的规定。如果没有明确的业务流程定义和有效的内部控制结构,企业组织将面临诉讼风险,甚至刑事处罚。与此同时,无论是欧盟的《General Data Protection Regulation》(GDPR,欧盟通用数据保护条例)[3],还是我国的《网络安全法》[4]、《民法典》[5]《数据安全法》[6]及《个人信息保护法》[7],都对企业的经营管理,尤其是对跨国跨境企业的数据流通经营管理提出了更高的合规性要求。然而另一方面,用于企业信息系统的大多数商品软件是基于"最佳实践"统一开发的,而特定企业组织信息处理流程会有个性化差异及要求;同时由于内外部环境影响,信息系统的变化速度跟不上企业的业务过程变化速度,以及企业组织中不同利益相关人员之间的需求有冲突等因素,导致信息系统和业务处理的实际过程、员工需求、管理之间频繁出现不匹配的情况,经常需要进行合规性检测。合规性检测就是要保障信息系统、业务过程和企业组织之间符合法律规章,通过分析实际过程并诊断差异,从中获得启发,以改进信息系统对业务流程的支持[8-9]。

基金项目:国家自然科学基金(61862047,62066034);内蒙古科技计划(201802028,2020GG0186)
This work was supported by the National Natural Science Foundation of China(61862047,62066034) and Inner Mongolia Science & Technology Plan(201802028,2020GG0186).
通信作者:李华(cslihua@imu.edu.cn)

Ramezani 等[10]确定了与合规性相关的 5 种类型的活动：合规性启发、合规性形式化、合规性实施、合规性检测和合规性改进。其中合规性检测可以是前向合规性检测(流程执行前)或后向合规性检测(流程执行后)。后向合规性检测主要是将事件日志中的事件与过程模型中的活动进行对比，旨在找到记录在日志中的实际情况和模型之间的吻合程度[8]。不同的方法考虑不同类型的逻辑模型，以形式化地捕获业务行为流程模型。合规性形式化可以是基于逻辑语言的(如 LTL)或者是基于流程模式(如 Petri-net)。合规性检测需求一般可以分为功能性属性和非功能性(或服务质量相关)属性[11]。目前对于合规性检测需求中功能性属性(如事件处理流程等方面)是否满足的研究中均基于模型及观测到行为的语义信息，以此对比两者之间的共性和差异，而缺乏对语义信息未知情形下的合规性检测研究。由于逻辑语言(如 LTL)不支持实时约束的规范[12]，传统 Petri 网原本概念排斥全局时钟[13]等因素，造成了合规性研究中对非功能性属性是否满足的研究相对较少。

然而由于政策或信息保护等，合规性检测面临语义信息未知的情况，如何在未知语义信息的情况下进行合规性检测是目前合规性研究中亟需解决的问题。与此同时，现在的各类系统的服务质量已经成为一个很重要的指标，合规性检测检验合规性需求中的非功能属性(如事件处理时间等方面)是否合规也成为了需要认真考虑的问题。

图同构是研究两个图是否具有完全相同的形式，被广泛应用于化学、生物信息、电子电路设计及计算机等领域中图结构的相似模式挖掘。由于子图同构合规性检测方法关注的是图的结构及图中点与边的连接关系，因此即使在不确定待检测模型语义信息的情况下，也可以通过对比图的结构以及图中点与边的连接关系，来达到将事件日志中的事件与过程模型中的活动进行对比的目的，找到记录在日志中的实际情况和建模行为之间的共性和差异，实现合规性检测。

本文的主要贡献如下：

1)定义基于图同构的合规性检测框架，通过可配置的子图同构合规性检测方法，将合规性检测问题转化为图同构问题。

2)检测模型语义信息不明情况下的合规性需求中的功能属性是否合规，即通过对比带标签的时间约束 Petri 网表示的规则模型的扩展可达图和含时间戳的事件日志记录的实际数据流通模型的共性与差异，来检测日志记录的事件处理流程是否合规。

3)利用扩展可达图中的旁标代表对应变迁的时间约束因素，检测合规性需求中的非功能属性是否合规。

本文第 1 章主要引出合规性检测的需求；第 2 章介绍文中用到的相关基本概念；第 3 章介绍基于图同构的合规性检测框架设计，并给出子图同构合规性检测算法；第 4 章通过实验分析介绍含时间戳的事件日志合规性检测方法的可行性；第 5 章介绍了相关工作；最后总结全文并展望未来。

2 预备知识

Petri 网[14]是一个建模分布式系统的数学工具，可以描述并发、非确定性、通信和同步的概念，一般表示为四元组 $N=(S,T,F,w)$，其中 S 是有限的库所集且 $S\neq\varnothing$，T 是有限的变迁集且 $T\neq\varnothing$，$F\subseteq(S\times T)\cup(T\times S)$ 是从库所到变迁和从变迁到库所的有向弧集合，$w:F\to\{1,2,3,\cdots\}$ 是弧上的权重函数。

时间 Petri 网(Timed Petri Nets，TPN)[15]对 Petri 网增加了时间，在变迁或库所中引入了相对时间因素，使得它们能够对系统行为和时间属性进行分析，多应用于对实时系统的分析。时间 Petri 网的典型代表有 Merlin 的时间 Petri 网等[16]。Merlin 将 TPN 定义为具有与每个变迁或库所相关联的时间间隔的网络，Tsai[17]在 TPN 的基础上提出了时间约束 Petri 网(Timing Constraint Petri Nets，TCPN)。为了表示事件处理动作，本文在时间 Petri 网的基础上，将带标签的时间约束 Petri 网定义如下。

定义 1(带标签的时间约束 Petri 网(Labeled Timing Constraint Petri Net，LTCPN)) 带标签的时间约束 Petri 网是一个九元组 $LTCPN=(S,T,F,w,A,l,C,D,M)$：$(S,T,F,w)$ 是 Petri 网结构；A 是事件处理动作的有限集合且 $A\neq\varnothing$；l 是变迁与事件处理过程的标记映射函数，$l:T\to A$；C 是整数对集，$(TC_{\min}(st_j),TC_{\max}(st_j))$ 是局部约束时间，其中 st_j 是库所或者变迁；D 是实数，$T_D(st_j)$ 是延迟的集合；M 是 m 个向量的标识集合，M_0 是 LTCPN 的初始令牌分布。

对于含有冲突结构的 Petri 网，变迁的触发具有不确定性，在分析其时间约束时，需将冲突结构分为若干个不含冲突结构的 T 网，再对其调度进行分析。文献[17-22]对时间 Petri 网的调度问题进行了讨论，时间约束 Petri 网中变迁的可触发时间段由令牌到达变迁的前驱库所时间及前驱库所的局部约束时间和变迁自身的局部约束时间共同决定，本文假设网中第一个变迁始终可触发且触发结束时间为 q_1，就时间约束 *Petri* 网中其余变迁的可触发时间段做如下定义。

定义 2(时间约束 Petri 网中变迁的可触发时间段[18]) 假设变迁 $t_j\in T$，库所 $s_j\in S$，如果 t_j 的所有前驱库所 s_j 具有使其触发所需的令牌，令牌到达 s_j 的时间为 $Tokarr(s_j)$，在 $Tokarr(s_j)+TC_{\min}(s_j)$ 到 $Tokarr(s_j)+TC_{\max}(s_j)$ 时间内，t_j 是使能的；$TF_{\min}(t_j)$ 和 $TF_{max}(t_j)$ 分别表示变迁 t_j 的最早可触发时间和最晚可触发时间，对于无冲突结构的变迁 t_j，如果 $TF_{\max}(t_j)-TF_{\min}(t_j)\geqslant t_d(t_j)$ 成立，其中

$$TF_{\min}(t_j)=\max\{\min\{Tokarr(s_j)\}+TC_{\min}(s_j)\}+TC_{\min}(t_j)$$

$$TF_{\max}(t_j)=\min\{\min\{\max\{Tokarr(s_j)\}+TC_{\max}(s_j)\},\max\{\max\{Tokarr(s_j)\}+TC_{\min}(s_j)\}+TC_{\max}(t_j)\}$$

则 t_j 是可触发的，记作 $t_j\in fireable(M)$。

定义 3(带标签的时间约束 Petri 网语义[19]) 一个带标签的时间约束 Petri 网 LTCPN 的语义为一个标签时间变迁系统 $LTTS_{\text{LTCPN}}=((F,O_1),l_t)$，其中 $O_1=(M_0,q_1)$，l_t 表示 t 触发时的标签，则：

$$\forall t\in T,(M,q)\xrightarrow{t,l_t}(M',q')\ \text{iff}$$

$$\begin{cases}t\in fireable(M),M'=M[t>,TF_{\min}(t)\leqslant q\leqslant TF_{\max}(t)\\ \forall t'\in T,q'=\begin{cases}q+t_D(t)+t_D(s'),t'\in fireable(M')\\0,\text{otherwise}\end{cases}\\ l_t=l((M,q)\xrightarrow{t}(M',q'))\end{cases}$$

其中，q 表示 t 的实际触发时间，M' 表示 t 触发后达到的后继

标识，记作 $M'=M[t>$，t' 表示 t 的后继变迁，s' 表示 t 的后继库所，q' 表示 t' 的实际触发时间，q_1 表示第一个变迁触发结束时间。

对 Petri 网的分析既可以通过可达树也可以通过可达图分析，Zhou 等[23]通过比较得出，可达图在分析 Petri 网的可达性上比可达树更优越，因此本文基于可达图进行合规性检测研究。可达图的构造有多种方法[24-25]，当节点对应可达标识，边对应变迁的发生，由节点和边及旁标的有向图组成了 Petri 网的可达图[13,26]。为了丰富可达图的表达，Zhang[27]等提出扩展可达图概念。

定义 4（扩展可达图） 假设 $LTCPN=(S,T,F,w,A,l,C,M)$ 是一个带标签的时间约束 Petri 网，LTCPN 的可达图可以定义为一个三元组 $RG(LTCPN)=(V(M_0),E,P)$，其中，$E=\{\langle M,M'\rangle \mid M,M'\in V(M_0),\exists t_k\in T:M[t_k>M'\}$，$P:E\to T$，$P(M,M')=t_k$ 当且仅当 $M[t_k>M'$，称 $V(M_0)$ 为 $RG(LTCPN)$ 的可达节点集，E 为 $RG(LTCPN)$ 的有向弧集；若 $P(M,M')=t_k$，则称 t_k 为弧 (M,M') 的旁标。

为了使读者更直观地了解 Petri 网及扩展可达图，现举一个简单的例子进行说明。

如图 1(a)为一个简单的 Petri 网，由 9 个库所和 8 个变迁组成，对应的流程由 $t1$ 开始，经过 $t2$，$t3$ 到 $t8$ 结束，或者经过 $t4$，之后 $t5$ 与 $t6$ 并行，经过 $t7$ 到达 $t8$ 结束。图 1(b)为其仿真运行后的可达图在旁标中的标注标签及可触发时间段的扩展可达图。

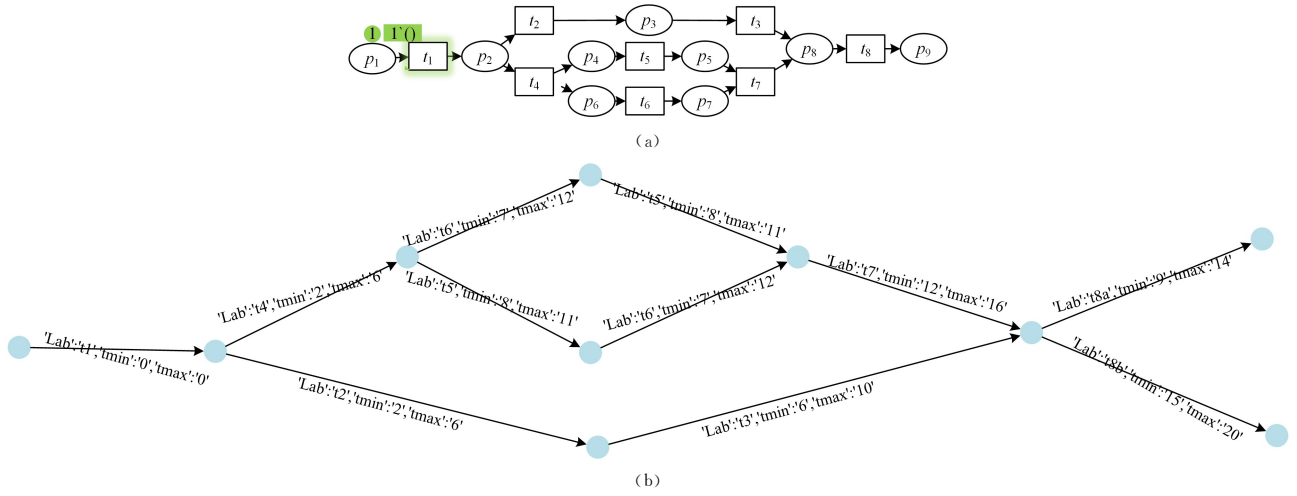


图 1 简单 Petri 网及其扩展可达图

Fig. 1 Simple Petri net and its extended reachability graph

Petri 网的扩展可达图是一个有向图，其中旁标可以代表弧的代价、事件处理过程、事件处理时间等，由于 CPN Tools[28]是一款用于编辑、模拟和分析着色 Petri 网的工具，可以生成完整的状态空间，以分析 Petri 网的有界性和可覆盖性等属性，因此本文采用 CPN Tools 建立 Petri 网模型，并获取其所有可能的可达图。通过旁标扩展可达图，达到合规性检测的要求。

事件日志是记录系统执行过程中发生的事件经过，本文从事件日志中提取事件处理过程模型，以此理解系统的活动过程。含时间戳的事件日志可以定义如下。

定义 5（含时间戳的事件日志[9]） 假设 A 是事件处理名称的有限集合，事件 $event\in A$，$\#_{time}(event)$ 是事件 $event$ 的时间戳；A^* 为 A 的有限序列集合，轨迹 $\sigma\in A^*$，则由 σ 组成的非空多集称为 A 上的事件日志 L，且对任意 i 和 j 满足 $1\leqslant i\leqslant j\leqslant|\sigma|:\#_{time}(\sigma(i))\leqslant\#_{time}(\sigma(j))$。

由于 Petri 网的可达图是有向图，可以用有向图及图的性质来对其进行分析，本文涉及的有向图及图的相关概念如下。

定义 6（有向图的通路[29]） 设 n 是非负整数且 $G(V,E)$ 表示有向图，在 G 中从顶点 u 到顶点 v 的长度为 n 的通路，由 G 的弧的序列 $e_1,e_2,\cdots,e_n$ 构成，使得 $e_1=(x_0,x_1)$，$e_2=(x_1,x_2)$，$\cdots$，$e_n=(x_{n-1},x_n)$，其中 $x_0=u$，$x_n=v$。

通常，把在相同的顶点上开始和结束的长度大于 0 的通路称为回路或圈。若两个图的顶点数、边数、顶点度及通路均相同，则可以认为这两个图同构。数学上，图的同构定义如下。

定义 7（图的同构[29]） 设 $G_1=(V_1,E_1)$ 和 $G_2=(V_2,E_2)$ 是图，其中 V_1 和 V_2、E_1 和 E_2 分别表示 G_1 和 G_2 的顶点集和弧集。若存在从 V_1 到 V_2 的一一对应函数 f，使得对所有属于 V_1 的 v_{1i} 和 v_{1j}，$\langle f(v_{1i}),f(v_{1j})\rangle\in E_2$ 当且仅当 $\langle v_{1i},v_{1j}\rangle\in E_1$，则 G_1 与 G_2 是同构的，函数 f 被称为同构（Isomorphism）。

同构的图都有或都没有的性质称为图同构不变量[29]。例如，有向图中顶点数、边数、顶点度及通路在同构的图中都应该是相同的，它们均属于有向图的图同构不变量。若两个有向图的图同构不变量相同，则这两个图是同构的。目前对图同构判定已经进行了一些研究[30-31]，虽然子图同构被认为是 NP-Complete 问题，但是 Petri 网的可达图是一种特殊类型的有向图，我们的目标是获得自动判断实际数据流通模型与规则模型扩展可达图中结构及属性相符部分的方法，因此其复杂度低于 NP-Complete。

3 基于图同构的合规性检测框架设计

van der Aalst[32]等对合规性的定义为：给定一个事件日志和一个 Petri 网，诊断观察到的行为（即事件日志中的痕迹）和建模行为（即 Petri 网的触发序列）之间的差异。但是其进行合规性检测时需要知道变迁的具体含义，不能处理变迁含义未知情况下的合规性检测，同时，其也未考虑非功能属性的合规性检测问题。本文提出的基于图同构的合规性检测框架，借助对比事件日志痕迹与扩展可达图结构的差异来达到判定合规性的目的，不仅可以处理已知语义信息甚至未知

语义信息的流程模型合规性检测问题，同时还可以通过扩展表示 Petri 网触发序列的可达图，丰富可达图的内涵，解决非功能属性的合规性检测问题。

3.1 问题定义

用 Petri 网对数据流通过程需要遵守的规章进行建模，记作基于规则的 Petri 网 NR。NR 中的标记表示请求的数据，相应类别的请求数据由对应的数据流通处理流程分别处理，表示该类别的数据流通过程中的流通规定，数据请求过程由开始库所运行到结束库所，表示该流程完结。由于待流通数据类别的差异，决定了其流通过程需要遵循不同的处理规则，对应于不同的处理流程。而 Petri 网的可达图是可达标记的分布，可达图可以充分反映 Petri 网的可达性、覆盖性等相关性质，并且，其边即为原 Petri 网中的变迁，代表数据流通处理流程中的处理过程；边上的约束为原 Petri 网中变迁触发时间的约束，代表相关流程的合规处理时间；边的连接关系，代表符合规章规定的处理过程模型下所有可能的数据流通处理过程的偏序关系。

数据流通系统中的日志记录了不同类型数据在流通过程中的痕迹，包含相应处理流程的处理记录及时间戳信息，根据对同一处理过程中不同处理流程序列轨迹 σ 可以构建出基于含时间戳的事件日志跟踪的实际数据流通模型，其边代表观察到的数据流通环节的实际处理过程，边上的时间表示观察到的该处理过程实际耗费的时间，边的连接关系代表观察到的实际数据流通环节处理过程的偏序关系。

因此，数据流通合规性检测问题可以转换为判断基于含时间戳的事件日志跟踪的实际数据流通模型与基于规则的 Petri 网 NR 扩展可达图之间的差异问题。如果能够在基于规则的 Petri 网 NR 扩展可达图中找出一个与基于含时间戳的事件日志跟踪的实际数据流通模型点与边连接结构的同构子图，且弧标签相同，处理时间在规则约束的时间范围内，则产生此数据流通事件日志的处理过程是合规的。

图 2 给出了本文合规性检测的思路。

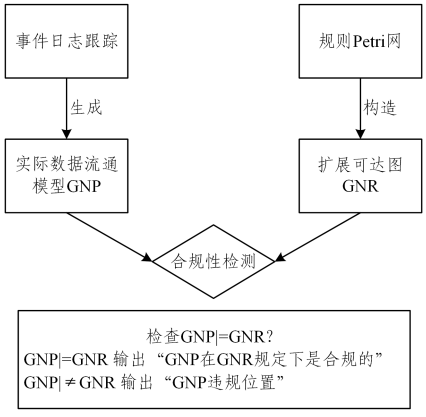


图 2 合规性检测方法

Fig. 2 Compliance check method

首先将基于含时间戳的事件日志的跟踪构造成实际数据流通模型 GNP，表示规则的 Petri 网模型生成扩展可达图 GNR，合规性检测过程通过检测 GNP⊧GNR 是否成立来判断基于含时间戳的事件日志跟踪的数据流通过程是否是符合基于规则的 Petri 网描述的规则要求。如果 GNP⊧GNR，表示在基于规则的 Petri 网 NR 扩展可达图 GNR 中至少存在一个与基于含时间戳的事件日志跟踪的实际数据流通模型 GNP 点与边连接结构的同构子图，且弧标签相同，处理时间在规则约束的时间范围内，则 GNP 在 GNR 规定下是合规的。如果 GNP |≠GNR，合规性检测程序会输出违规的事件处理过程。

3.2 框架设计

针对图 2 的合规性检测方法，基于图同构的合规性检测框架设计如下：

步骤 1 构造描述规则的 Petri 网扩展可达图 GNR；

步骤 2 生成事件日志跟踪的实际数据流通模型 GNP；

步骤 3 分析 GNR 和 GNP 图结构及对应弧属性是否同构，若同构则输出合规；否则输出差异位置。

1)构造扩展可达图

由于普通的 Petri 网中，只要变迁的所有输入库所具有令牌，该变迁即是可触发的，但根据定义 2，在时间约束 Petri 网中，变迁的触发还需要满足时间约束条件限制，因此在普通 Petri 网中可达的状态，在时间约束 Petri 网中则不一定可达[17-18]。为了得到全部状态转移过程的可达图，本文在利用现有建模工具生成可达图时，而是不考虑时间约束规则，先生成一个包含全部状态转移过程的可达图。

由定义 2 可知，在时间约束 Petri 网中变迁的可触发时间段与令牌到达该变迁的前驱库所时间，该变迁触发前等待的时间及该变迁的前驱库所支持后续变迁使能的时间有关，因此，在计算时间约束 Petri 网中变迁的可触发时间段时，需首先获取令牌到达该变迁的前驱库所时间，而该变迁的可达路径中记录着令牌经过的库所及变迁的先后顺序，从而可以得到令牌到达该变迁的前驱库所时间，并进一步得出时间约束 Petri 网中变迁的可触发时间段。

算法 1 在普通 Petri 网可达图基础上，通过可达路径，根据定义 2 和定义 3 中规定的变迁可触发时间段及 Petri 网语义，构造符合时间约束 Petri 网的扩展可达图。

算法 1 扩展可达图构造算法

输入：Petri 网可达图 SNR，Petri 网模型 NR

输出：扩展可达图 GNR

BEGIN

1. 初始化 GNR←SNR；
2. 遍历 GNR 获得所有可达路径集 pathlist；
3. for each path in pathlist
4. for each node in path
5. 获得 node 在 NR 中对应的库所列表 slist；
6. 根据定义 3 计算 Tokarr(sj)；
7. if path 中存在从 node 发出的边 arc
8. 获得 arc 在 NR 中对应的变迁 t；
9. 根据定义 2 计算$[TF_{min}(t), TF_{max}(t)]$；
10. if arc 在 GNR 中对应的弧没有关联标签及可触发时间
11. 为 arc 在 GNR 所对应的弧关联标签及$[TF_{min}(t), TF_{max}(t)]$；
12. END if
13. else
14. if node 有前驱节点在 GNR 中存在多个入弧
15. GNR 添加新节点与弧并为弧关联新的标签及$[TF_{min}(t), TF_{max}(t)]$；
16. END if
17. END else

```
18.      END if
19.    END for
20. END for
21. return GNR;
END
```

通过算法 1 对普通 Petri 网可达图进行扩展，生成带标签的时间约束 Petri 网的扩展可达图。算法 1 的输入为 CPN Tools 生成的普通 Petri 网可达图，其输出为带标签的及带时间约束的扩展可达图。由 Petri 网可达图的性质可知，可达路径集 *pathlist* 中的可达路径已经将冲突的变迁结构分解，可达路径对应的变迁发生序列不包含冲突结构，因此算法 1 第 2 行得到的可达路径集中的变迁触发序列符合定义 2 规定的前提条件。算法 1 第 11 行为待处理的扩展可达图弧表示的变迁关联对应标签，并关联可触发时间，即：A 是事件活动的有限集合，旁标标签函数 $P_{NRl}:E_{NR}\rightarrow A_{NR}$ 为 GNR 的弧标签；$[TF_{\min}(t_j),TF_{\max}(t_j)]$表示变迁可触发时间段，旁标时间函数 $P_{NR\lambda}:E_{NR}\rightarrow[TF_{\min}(t_j),TF_{\max}(t_j)]$为 GNR 的弧，表示变迁可触发的时间段。由于令牌在经过 Petri 网的冲突结构后，在汇聚节点处可能会产生多个令牌到达时间，因此算法 1 第 14 行对冲突结构的汇聚路径进行判定，第 15 行对汇聚路径添加新的标签及可触发时间。

2）生成实际数据流通模型

含时间戳的事件日志跟踪是系统实际运行状态的记录，详细记录事件实际发生的全过程。通常情况下，将包含一个事件中所有活动的名称及活动开始时间和结束时间等相关信息提取到含时间戳的事件日志序列表中，由事件日志跟踪序列根据算法 2 生成含时间戳的事件日志跟踪序列的实际数据流通模型。

算法 2　含时间戳的事件日志跟踪序列的实际数据流通模型生成算法

```
输入：含时间戳的事件日志序列表
输出：实际数据流通模型 GNP
BEGIN
1. 定义一个空的数据公开列表
2. 遍历含时间戳的事件日志序列表表中的每个实例对象
3.    初始化一个空的实际数据流通模型
4.    获取每个实例对象的行数以及列数
5.    if 行数>0 and 列数>0
6.       遍历实例对象的每一行
7.          通过从每一行获取的活动信息，构造实际数据流通模型的
            一条弧
8.       END 遍历
9.    END if
10.   将此时构造的实际数据流通模型图添加到数据公开列表中
11. END 遍历
12. return 实际数据流通模型 GNP
END
```

通过算法 2 生成含时间戳的事件日志跟踪序列的实际数据流通模型。算法 2 的第 7 行分别用对应的事件处理名称给弧的标签赋值，用事件实际处理时间给弧的时间赋值，即：$[T_{\text{start}}(t_j),T_{\text{complete}}(t_j)]$表示变迁实际执行时间段，标签函数 $P_{NPl}:E_{NP}\rightarrow A_{NP}$ 为 GNP 的弧标签，时间函数 $P_{NP\lambda}:E_{NP}\rightarrow[T_{\text{start}}(t_j),T_{\text{complete}}(t_j)]$为 GNP 的弧，表示变迁实际执行的时间段。

3）合规性判定

定义 8（含时间戳的事件日志跟踪序列合规性）　假设基于含时间戳的事件日志跟踪序列获得的实际数据流通模型为 $GNP=(V_{NP},E_{NP},P_{NP})$，基于规则的 Petri 网 NR 的扩展可达图为 $GNR=(V_{NR},E_{NR},P_{NR})$，$V_{NP}$ 和 V_{NR}、E_{NP} 和 E_{NR}、P_{NP} 和 P_{NR} 分别表示 GNP 和 GNR 的顶点集、弧集和旁标集；$f(u)$表示 GNR 中与 $u\in GNP$ 相关联的顶点，$l_{NP}(u,u')$和 $l_{NR}(f(u),f(u'))$ 分别表示 GNP 和 GNR 中弧的标签，$\lambda_{NP}(u,u')$和 $\lambda_{NR}(f(u),f(u'))$分别表示 GNP 和 GNR 中弧对应的变迁触发和执行的时间段。GNP 在 GNR 规定下是合规的，即基于此事件日志的案例流程是合规的当且仅当 GNR 中至少存在一个部分结构是 GNP 的同构子图且弧含义相符。

其中，同构子图指存在 GNP 和 GNR 顶点之间的一一映射构成的集合 $Z\subset V_{NP}\times V_{NR}$，且满足：

1）$\forall u\in V_{NP},\exists f(u)\in V_{NR}:(u,f(u))\in Z$

2）$\forall u,u'\in V_{NP},u\neq u'\Rightarrow f(u)\neq f(u')$

3）$\forall (u,u')\in E_{NP},\exists (f(u),f(u'))\in E_{NR}$

4）$\forall (u,u')\in E_{NP},l_{NP}(u,u')=l_{NR}(f(u),f(u'))$

5）$\forall (u,u')\in E_{NP},\lambda_{NP}(u,u')\ in\ \lambda_{NR}(f(u),f(u'))$

条件 1），2），3）表示两个图的节点和边连接结构相符，条件 4）和 5）表示弧含义相符。

GNR 中边的连接关系代表符合规章规定的处理过程模型下所有可能的数据流通处理过程的偏序关系，边标签及约束代表数据流通处理流程中的处理过程及相关流程的合规处理时间。GNR 中至少存在一个部分是 GNP 的同构子图，表示在规则模型中存在一个实际数据流通处理过程的偏序关系相同的部分，则实际数据流通处理过程的结构是合规的。GNP 中每个变迁实际执行时间段在 GNR 相应的变迁可触发的时间段内，表示此处理流程结构中的每个处理事件及对应处理事件的时间约束是合规的。

算法 3　获取匹配路径算法

```
输入：a_GNR，a_GNP，GNR，GNP，tmp，compare
输出：匹配路径 matchPath
BEGIN
1. if a_GNR与 a_GNP的 compare 符合其相应的规则
2.    matchPath. add((a_GNR，a_GNP));
3.    获得 a_GNR，a_GNP的后继边的列表 a_GNR_out，a_GNP_out;
4.    for each arc1 in a_GNR_out
5.       for each arc2 in a_GNP_out
6.          if arc2不属于 tmp
7.          调用算法 3(arc1，arc2，tmp，GNR，GNP，compare);
8.          END if
9.       END for
10.   END for
11. END if
12. else
13.   return matchPath;
14. END else
END
```

算法 4　合规性检测算法

```
输入：GNR，GNP
输出：合规性检测成功部分
BEGIN
1. 获取 GNR 从开始节点发出的边 arcGNR;
```

2. 获取 GNP 从开始节点发出的边 arc_{GNP};
3. 初始化 tmp 为空;
4. if 调用算法 3(arc_{GNR}, arc_{GNP}, GNR, GNP, tmp, struct)获得 matchPath 不为空
5. if 调用算法 3(arc_{GNR}, arc_{GNP}, GNR, GNP, tmp, lab)获得 matchPath 不为空
6. if 调用算法 3(arc_{GNR}, arc_{GNP}, GNR, GNP, tmp, [TF_{min}, TF_{max}])获得 matchPath 不为空
7. return "合规";
8. END if
9. else
10. return "流程合规,变迁时间开销不符,事件处理时间不合规";
11. END else
12. END if
13. else
14. return "流程不合规";
15. END else
16. END if
17. else
18. return "流程结构不合规";
19. END else
END

其中,算法 4 第 4—6 行通过调用算法 3 的获取匹配路径算法,分别从两个图的结构、图中弧的标签及弧所对应的变迁可触发时间等方面进行合规性检测。合规性判定的依据是判断两个图的点与边连接结构和弧含义两个部分,特别是在弧含义未知的情况下,可以只通过判断 GNR 与 GNP 中是否有部分一致的结构来判定 GNP 中处理流程是否合规。算法 3 通过递归方式检测 GNR 中是否存在一个部分与 GNP 是图点与边连接结构的同构子图且弧含义相符,如果 GNP 全部属于合规性检测成功部分,则 $GNP \vdash GNR$ 成立,GNP 在 GNR 规定下是合规的;否则,$GNP \notin$ 合规检测成功部分的过程即为违规的过程。

3.3 方法分析

本文提出的基于图同构的合规性检测框架,不仅适用于处理语义信息不明的流程模型合规性检测问题,同时还适用于处理已知语义信息的流程模型非功能属性合规性检测问题。对于无任何语义信息的模型,基于图同构的合规性检测框架也可以通过节点与边的连接结构进行模型匹配,以完成流程的合规性检测;而已知语义信息的流程模型在应用基于图同构的合规性检测框架进行合规性检测时,节点或边的语义信息可以有效减少检测过程探索的状态空间数量,从而快速得出合规性检测结果。同时,本文还关注了事件处理时效相关要求,通过带标签的时间约束 Petri 网模型表示事件合规处理时间,对应于扩展可达图相应位置的旁标,在合规性检测过程中同时判定图点与边连接结构及对应位置弧属性是否合规,以此来检验事件处理流程及事件处理时间等方面是否合规。

根据计算复杂性理论[33],对算法的性能分析主要从算法的空间复杂度和时间复杂度两个方面进行分析。本文对于所设计的合规性检测框架的性能分析也从合规性检测算法的空间复杂度和时间复杂度两个方面进行分析。

合规性检测算法的空间复杂度主要由查找空间的规模决定。本文设计的合规性检测算法主要通过检测规则模型中是否存在一个与过程模型是同构子图的部分且弧含义相符,来完成合规性检测,其查找空间的规模仅取决于描述规则的模型,其复杂度也仅取决于描述规则模型的复杂程度。

合规性检测算法的时间复杂度主要由其核心比较部分算法的时间复杂度决定。本文设计的合规性检测算法,时间复杂度取决于探索两个图中的同构子图的部分且弧含义相符,已知的一般的无向图同构算法在最坏情形下的时间复杂度为 $O(N!)$[30],但是本文中所讨论的图为有固定起点和终点的有向图,同时图中的弧拥有不同的含义,因此可以有效地缩小探索空间,其最坏情况下的时间复杂度远小于 $O(N!)$。

4 在实际业务中的应用

本章通过数据申请系统事件日志事例,利用第 3 章中提出的方法,检验实际业务系统的合规性。同时,通过不同测试用例来测试方法的可行性和有效性。

本文的所有实验均在配置有 Intel(R) Core(TM) i5-9400F 的 CPU、32GB 内存、Windows 10 的操作系统的台式机平台上完成,Python 版本为 3.7。

4.1 实验背景

根据《中华人民共和国个人信息保护法》《中华人民共和国政府信息公开条例》等相关的法律法规对数据开放服务的法律规定,某市政务数据开放服务流程共分 3 个阶段:申请受理与分类处理阶段、异议解决阶段、检查复评阶段。

申请受理与分类处理阶段流程。申请方下载申请材料并且提交,受理方进行审查,并且在 7 个工作日内判断是否受理,如果不予受理,将向申请方发送《不予受理通知书》或《一次性补正通知书》;若受理,将向申请方发送《正式受理通知书》并会依不同情形分类处理:1)若申请方申请已汇聚无条件开放的数据则直接当场答复申请方通过某市政务数据资源网在线获取使用;2)若申请方申请金融数据专区具备开放开放条件的数据则直接当场答复申请方到某市首贷服务中心获取使用并且将申请材料移交至金融数据专区管理方;3)若申请方申请尚未汇聚使用频次可能偏低的数据,则当场答复申请方相关数据所有单位申请的渠道和方式方法;4)若申请方申请尚未汇聚使用频次可能偏高的无条件开放数据,则进行数据汇总并且启动数据汇聚工作,完成后答复申请方获取渠道;5)若申请方申请已汇聚并涉及数据源提供部门有条件开放的数据,则移交材料并详细审查以及依据相关规定答复申请方;6)若申请方申请尚未汇聚使用频次可能偏高的有条件开放数据,则进行数据汇总、启动数据汇聚工作、移交材料并详细审查以及依据相关规定答复申请方。对于前 4 种情形,申请方接收正式答复信息同时流程结束;对于后 2 种情形,受理方判断是否属于 3 种不予开放的情况。如若属于不予开放的情况,受理方会同数据资源提供部门向申请方发放《不予开放通知书》,同时流程结束;如若不属于,受理方会同数据资源提供部门要求申请方签署《承诺函》,受理方备案,申请方接收所申请的数据,流程结束。其中,3 种不予开放的情况:1)涉及国家秘密、商业秘密、个人隐私,或者法律法规不得开放得数据;2)申请方资质、数据用途、数据处理和安全管理能力等不符合要求;3)第三方相关机构不同意开放。

异议解决阶段流程。申请方对数据开放的答复提出

异议，向某市政府信息公开主管理部门申请协调解决；某市政府信息公开主管理部门接收异议申请并协调解决给出答复，流程结束。

检查复评阶段流程。数据源提供部门以及受理方定期对申请方进行检查复评，若不符合要求，向申请方发送《数据开放服务终止通知》；申请方接收《数据开放服务终止通知》并反馈信息；受理方及数据源提供部门接收反馈并备案，流程结束。

4.2 构造描述规则的 Petri 网扩展可达图 GNR

对上述数据开放服务流程规则进行分析，建立一个描述数据开放服务流程规则的 Petri 网模型 NR，为了方便分析，添加相关辅助变迁，用 CPN Tools 对其进行建模，如图 3 所示。

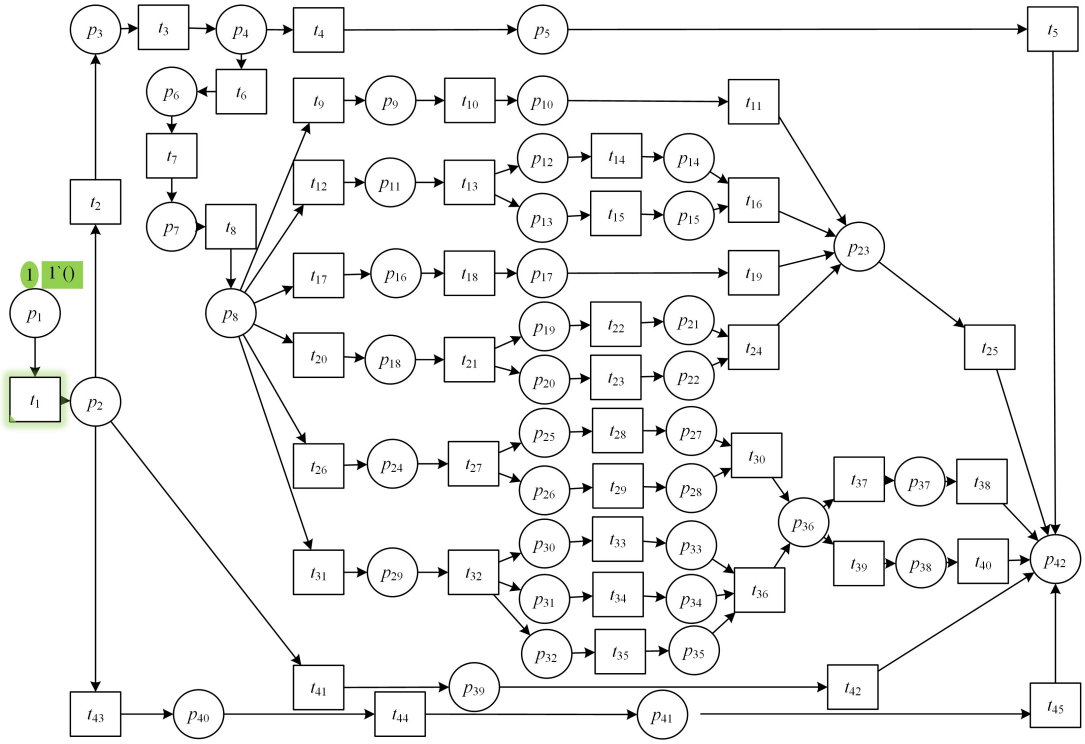


图 3 CPN Tools 建模描述规则的 Petri 网模型 NR

Fig. 3 CPN Tools modeling rule description Petri net model NR

CPN Tools 仿真运行后，生成其可达图，如图 4 所示。根据 3.2 节所提的算法 1，分别分析仿真运行后的可达图中的可达路径及每个变迁的可触发时间段，并在可达图的旁标中标注标签及可触发时间段，构造描述规则的 Petri 网模型 NR 的扩展可达图，其扩展可达图的局部如图 5 所示。

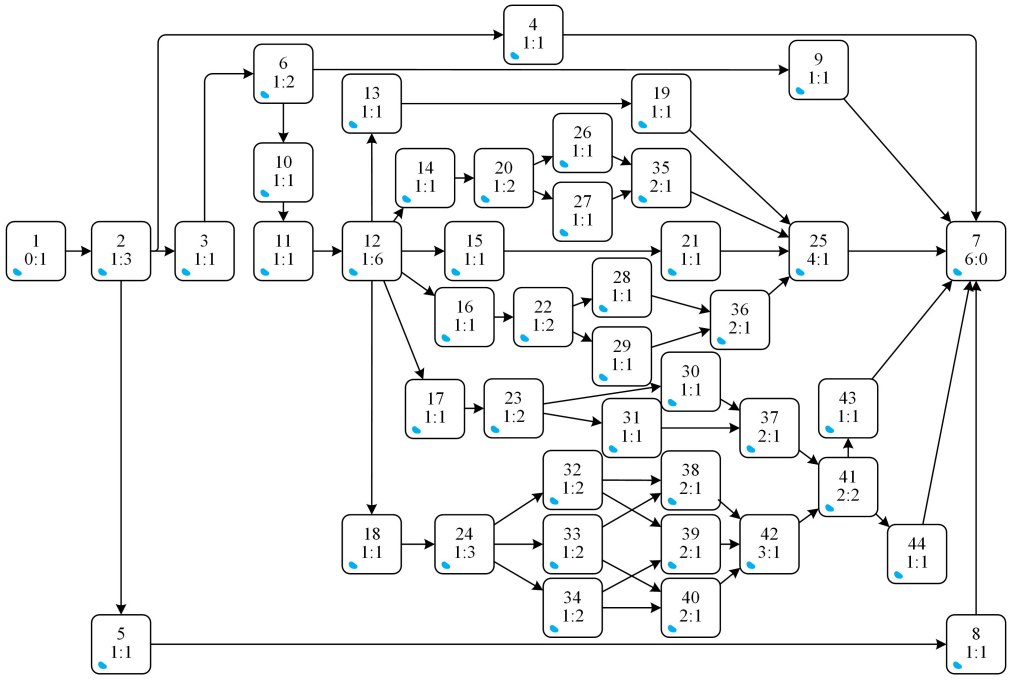


图 4 CPN Tools 仿真运行后的可达图

Fig. 4 Reachability diagram after CPN Tools simulation

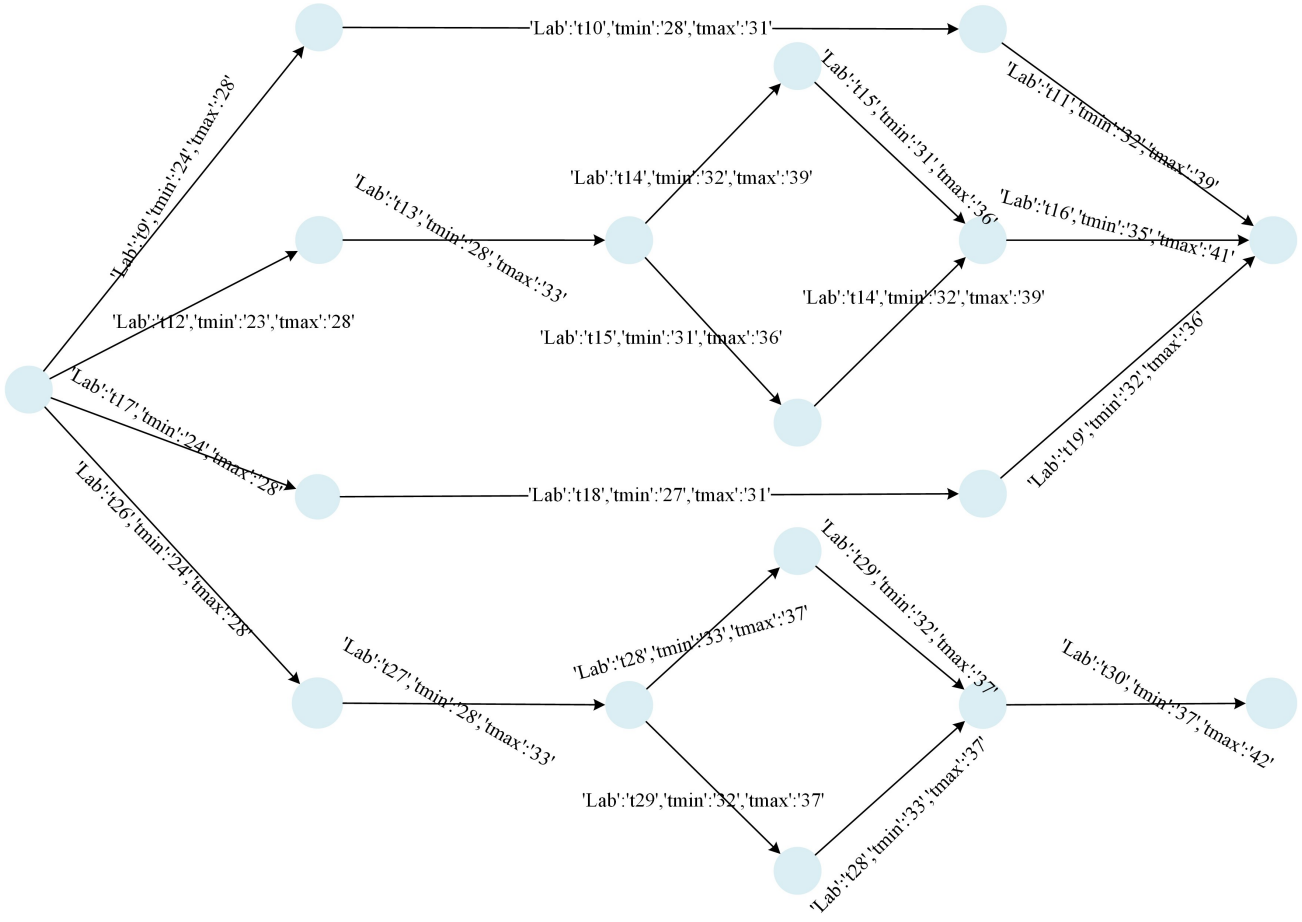


图 5 描述规则的 Petri 网模型 NR 的扩展可达图 GNR 的局部结构

Fig. 5 Local structure of extended reachability graph GNR of rule description Petri net model NR

描述规则的 Petri 网模型 NR 中每个变迁所代表的含义及其映射到的活动标签和每个变迁可触发时间段的部分变迁如表 1 所列，其中辅助变迁由框线表示。

表 1 描述规则的 Petri 网模型 NR 中部分变迁可触发时间段及其含义

Table 1 Partial transitions trigger time periods and their meanings ruledescription Petri net model NR

变迁	最早可触发时间	最晚可触发时间	含义
			…
$t10$	28	31	当场答复申请方在线获取使用
$t11$	32	39	完成要求工作流程
$t12$	23	28	申请方申请金融数据专区具备开放开放条件的数据
$t13$	28	33	完成以下流程
$t14$	32	39	当场答复申请方到首贷服务中心获取使用
$t15$	31	39	将申请材料移交至金融数据专区管理方
$t16$	35	41	完成要求工作流程
$t17$	24	28	申请方申请尚未汇聚使用频次可能偏低的数据
			…

规则模型 $NR=(SR,TR,FR,TF_{\min}R,TF_{\max}R)$ 如图 3 所示，其中，库所集合 $SR=\{p1,p2,\cdots,p42\}$，变迁集合 $TR=\{t1,t2,\cdots,t45\}$，流关系集合 $FR=\{(p1,t1),(t1,p2),(p2,t2),(p2,t41),(p2,t43),\cdots,(t5,p42),(t25,p42),(t38,p42),(t40,p42),(t42,p42),(t45,p42)\}$；模型中各变迁可触发时间段为 $TF_{\min}R(t1)=0$，$TF_{\max}R(t1)=0$，$TC_{\min}R(t2)=2$，$TC_{\max}R(t2)=6$，$TC_{\min}R(t3)=6$，$TC_{\max}R(t3)=10$，…，$TC_{\min}R(t45)=10$，$TC_{\max}R(t45)=16$，其中，对于受理无条件开放的数据的 4 种情形，有 4 条不同的处理路径，彼此之间相互冲突，令牌到达 $p23$ 时会产生 4 个不同的时间，因此在计算 $t25$ 的可触发时间段之前，通过可达路径集分解冲突，分别分析 4 种情况，此时，$t25$ 拥有 4 个可触发时间段，用 $t25(a)$，$t25(b)$，$t25(c)$，$t25(d)$ 区分；同理，对于受理有条件开放的数据的两种情形，彼此之间也为冲突结构，令牌到达 $p36$ 时会产生两个不同的时间，分解冲突后分别分析，因此 $t37$ 及 $t39$ 分别具有两个可触发时间段，用 $t37(a)$，$t37(b)$ 及 $t39(a)$，$t39(b)$ 区分。该规则模型是一个带标签的时间约束 Petri 网，具有安全性、无死锁与陷阱等性质，因此，该模型是一个合理的规则流程模型。

4.3 构造事件日志跟踪的实际数据流通模型 GNP

将系统中记录的与此示例对应的部分事件日志处理后如表 2 所列，每一行代表一个事件，事件已经按照不同的案例进行分组，并且为了便于分析，添加相应辅助变迁事件，由框线表示，比如案例 1 由 14 个相关事件和 4 个辅助变迁事件组成。每个事件都由一个规则模型中变迁对应的事件 ID 和记录了事件的相对开始时间与结束时间等信息组成。案例 2－4 中只标识出与案例 1 相异的部分事件。

表 2　数据开放服务系统事件日志

Table 2　Event log of data open service system

案例 ID	事件 ID	开始时间	结束时间	活动	…
1	$t1$	0	0	开始变迁	…
	$t2$	4	6	申请者下载申请材料并提交	…
	$t3$	6	8	受理方审查	…
	$t6$	9	13	受理	…
	$t7$	14	16	发送及接收《正式受理通知书》	…
	$t8$	20	22	受理方依不同情形分类处理	…
	$t31$	23	25	申请方申请尚未汇聚使用频次可能偏高的有条件开放数据	…
	$t32$	28	30	完成以下流程	…
	$t33$	30	32	数据汇总，启动数据汇聚工作	…
	$t35$	32	33	依据相关规定答复申请方	…
	$t34$	34	36	移交材料并详细审查	…
	$t36$	37	39	完成要求工作流程	…
	$t37$	41	42	属于 3 种不予开放的情况	…
	$t38$	45	48	发放并接收《不予开放通知书》	…
	$t9$	24	26	若申请方申请已汇聚无条件开放的数据	…
	$t10$	28	30	当场答复申请方在线获取使用	…
	$t11$	33	35	完成要求工作流程	…
	$t25$	38	40	申请方接收正式答复信息	…
				…	
2	$t34$	34	36(40)	移交材料并详细审查	…
				…	
				…	
3	$t34$	34	36	移交材料并详细审查	…
	$t36$	37	39	完成要求工作流程	…
	$t40$	45	48	~~申请方签署《承诺函》并接收所申请的数据~~	…
	$t9$	24	26	若申请方申请已汇聚无条件开放的数据	…
				…	
				…	
4	$t8$	20	22	受理方依不同情形分类处理	…
	$t18$	22	23	当场答复申请方相关数据所有单位申请的渠道和方式方法	…
	$t12$	23	25	申请方申请金融数据专区具备开放开放条件的数据	…
	$t13$	28	30	完成以下流程	…
	$t15$	30	32	当场答复申请方到首贷服务中心获取使用	…
	$t14$	32	33	将申请材料移交至金融数据专区管理方	…
	$t16$	34	36	完成要求工作流程	…
	$t39$	37	39	不属于 3 种不予开放的情况	…
	$t40$	45	48	申请方签署《承诺函》并接收所申请的数据	…
	$t9$	24	26	若申请方申请已汇聚无条件开放的数据	…
				…	

对于事件日志轨迹 σ_1，利用算法 2 构造其实际数据的　　数据开放服务模型，如图 6 所示。

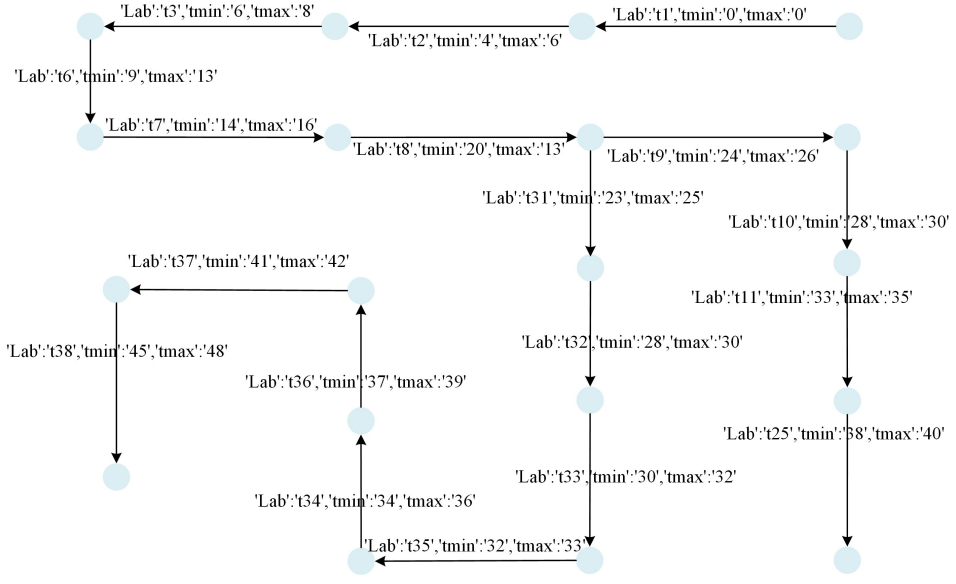


图 6　事件日志轨迹 σ_1 的实际数据流通模型 GNP σ_1

Fig. 6　Actual data flow model GNP σ_1 for event log trace σ_1

图 6 是根据事件日志轨迹 σ_1 构造出的实际数据流通模型 GNPσ_1，其中连接节点的边代表该案例中包含的事件处理过程，边的先后顺序代表实际事件处理过程的偏序关系，边上的标签代表实际处理事件 ID，时间表示实际事件处理开始时间和结束时间。

4.4 事件合规性检测的效果评估

数据流通的合规性检测本质即为判断表示实际数据流通过程的实际数据流通模型 GNPσ_1 是否为表示规则模型 NR 的扩展可达图 GNR 的子图，若 GNPσ_1 为 GNR 的子图，则实际数据流通过程是符合数据开放服务流程规定的，否则存在违规的过程。

为了充分测试合规性检测方法的可行性和有效性，对于 σ_1 保持为事件日志系统中的原始记录，对于其余事件日志 σ_{2-4} 则分别随机对日志中轨迹进行增删改操作，如遍历每条日志轨迹并对每个活动随机应用以下任意一种操作：1）在该活动前添加 1 个活动；2）修改该活动的名称；3）删除该活动；4）修改该活动开始时间；5）修改该活动结束时间；6）保持不变。对 σ_{2-4} 所做的改动标注在表 2 中，其中，单下划线处修改活动结束时间，删除线表示删除该活动，单下波浪线表示添加的活动。然后对日志序列进行合规性检测，以验证方法的可行性和有效性。

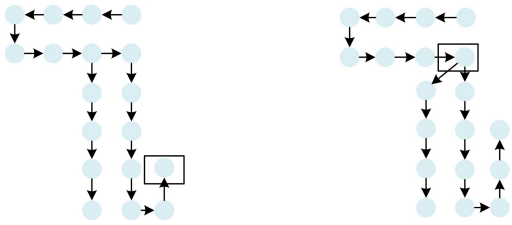
应用本文所提的合规性检测框架，对事件日志轨迹 σ_{1-4} 进行合规性分析，可以得到其实际数据的数据开放服务模型 GNPσ_{1-4} 中的合规性检测结果。根据定义 8，V 与 E 元组代表图的结构信息，因此，即使在 P 元组信息（边所代表的具体含义）未知的情况下，仅通过比较图的结构信息，也可以进行合规性检测，GNPσ_{1-4} 的事件处理流程合规性检测结果如表 3 所列。

表 3 事件处理流程合规性检测结果

Table 3 Detection results of event processing flow compliance

实际数据的数据开放服务模型	流程合规性检测结果
GNPσ_1	{([c1],[p1]),([c2],[p2]),([c5],[p3]),([c8],[p4]),([c9],[p6]),([c11],[p7]),([c12],[p8]),([c13],[p29]),([c19],[p30,p31,p32]),([c27],[p33,p31,p32]),([c36],[p33,p35,p31]),([c41],[p33,p34,p35]),([c42],[p36]),([c44],[p37]),([c45],[p42]),([c18],[p9]),([c24],[p10]),([c32],[p23]),([c46],[p42])}
GNPσ_2	{([c1],[p1]),([c2],[p2]),([c5],[p3]),([c8],[p4]),([c9],[p6]),([c11],[p7]),([c12],[p8]),([c13],[p29]),([c19],[p30,p31,p32]),([c27],[p33,p31,p32]),([c36],[p33,p35,p31]),([c41],[p33,p34,p35]),([c42],[p36]),([c44],[p37]),([c45],[p42]),([c18],[p9]),([c24],[p10]),([c32],[p23]),([c46],[p42])}
GNPσ_3	流程不合规
GNPσ_4	流程不合规

由表 3 及算法 4 分析可知，描述规则的 Petri 网模型 NR 的扩展可达图 GNR 中均有子图与事件日志轨迹 σ_1 和 σ_2 的实际数据的数据开放服务模型 GNPσ_1 和 GNPσ_2 是同构子图，而 GNPσ_3 与 GNPσ_4 均存在 GNR 中没有与之相对应的变迁结构，如图 7 所示，因此没有与事件日志轨迹 σ_3 和 σ_4 的实际数据的数据开放服务模型 GNPσ_3 和 GNPσ_4 是同构子图的部分，因此，在无法获取事件处理具体信息的情况下，仅通过 GNR 与 GNP 中点与边的结构特征，也可以进行合规性检测。



(a)GNPσ_3 (b)GNPσ_4

注：事件日志轨迹 σ_3 与 σ_4 的实际数据流通模型 GNPσ_3 与 GNPσ_4 不合规位置如图中黑框标出。

图 7 流程不合规示意图

Fig. 7 Schematic diagram of process noncompliance

定义 8 中 P 元组代表图中边对应的活动名称信息及相应的处理时间信息，进一步的检测结果如表 4 所列。由表 4 及算法 4 分析可知，虽然描述规则的 Petri 网模型 NR 的扩展可达图 GNR 与事件日志轨迹 σ_1 和 σ_2 的实际数据的数据开放服务模型 GNPσ_1 和 GNPσ_2 的弧含义相符，但是只有 GNPσ_1 中的每个事件处理时间均在相应的 GNR 中事件处理时间规定的范围内，而 GNPσ_2 中的 $GNP\sigma_2.t34.P\lambda=[34,40]$，而 $GNR.t34.P\lambda=[30,36]$，也就是说在实际变迁 $t34$ 的最晚触发时间大于规定的最晚可触发时间，GNPσ_2 中的事件处理时间超出了 GNR 中相应位置规定的最大处理时间，因此，事件日志轨迹 σ_1 的事件处理流程合规，事件日志轨迹 σ_2 的事件处理流程不合规。

表 4 事件处理非功能属性合规性检测结果

Table 4 Compliance check results of non-functional attributes of event processing

实际数据的数据开放服务模型	合规性检测结果
GNPσ_1	合规
GNPσ_2	流程合规，变迁时间开销不符，事件处理时间不合规

由表 3 及表 4 可以看出，利用算法 4 合规性检测算法可以有效识别出含时间戳事件日志跟踪中事件处理流程不合规及事件处理时间不合规等现象，对于与模型相符的事件日志可以正确判定为合规的事件处理流程，由此可见本文中设计的合规性检测方法是可行和有效的。

5 相关工作

形式化方法是保证计算机科学软硬件发展正确性、可靠性和安全性的重要方法[34]。常用的合规性检测方法中，将规则约束用线性时序逻辑（LTL）[35-36]表示，通过 LTL 检查器对比事件日志与约束的差异。但是 LTL 检查器在检测到第一个偏差时会丢弃其余的日志跟踪，并且不提供详细的诊断过程，因此不能检测出所有可能的违规行为。

El Gammal 等[12]中利用逻辑语言形式化表示流程过程的结构约束，利用相关的自动验证和分析工具来进行合规性验证和分析。Ramezani 等[37]基于 Petri 网形式化表达了 15 个类别的 55 个面向控制流的合规性规则，并利用对齐的方法对

合规性进行检查。但是,合规性检测可能需要定义更多的流程属性,例如与时间相关的约束,但是他们的研究并没有这方面的体现。

虽然 Van der Aalst 等[38-39]在传统 Petri 网基础上扩展了对时间的表示,这与本文研究的合规性形式化表示方式有相似之处,但是其侧重点在基于过去的事件模型实现对未来的预测,与本文研究的合规性检测目标是不同的。

Van der Aalst 等[8-9,40]指出合规性检查将事件日志中的事件与过程模型中的活动关联,并且将二者进行对比,旨在找到观察行为和建模行为之间的共性和差异。但是其基于简单事件日志模型,忽略了案例的相关属性,因此不支持对含有时间约束的规则进行合规性检测。

Adriansyah 等[41-43]基于将观察到的事件日志中的事件与过程模型中的活动进行对齐的思想实现合规性检查,其关注点在于定义一个适当的代价函数,引入 A^* 算法寻找多种对齐模式中的最优对齐,以此衡量事件日志中的系统运行行为和过程模型的一致性,在模型一致性的基础上,通过重放等技术进行合规性检查。但是这些方法存在查找空间过大,算法复杂性较高等问题。Han 等[40]通过将两个 Petri 网相乘,将日志模型与过程模型之间的拟合关系全部体现在一个 Petri 网中,从而将迹与过程模型之间的对齐运算转换成求 Petri 网可达状态的运算,但是,该方法存在空间浪费、查找时间复杂度高等问题。并且,无论是基于重放的技术还是基于对齐的方法,均无法判断事件过程未知情况下的合规性判断问题。本文所提方法可以仅通过判断图中点与边连接结构完成对事件过程未知情况下的合规性判断问题。

部分对业务流程合规性的研究基于业务流程建模与标注(BPMN)[44-45],BPMN 通过增加定义选择网关或者并行网关等符号元素来处理选择或者并发等不同业务流程过程,而 Petri 网自身所定义的库所与变迁连接顺序及变迁触发规则自然就涵盖了不同流程之间的选择或者并发关系,无须做特殊定义,同时,Petri 网完备的理论基础及丰富的分析方法更有利于进行合规性检测。

为了实现本文所提的合规性检测算法,我们还需要研究当前的图同构判定算法。图同构判定算法[30](VF2 算法)通过在两个图中进行遍历比较,以判断是否存在两个图中结构相同部分的映射函数关系。本文的算法可以认为是对图同构算法的扩展,本文利用图遍历和比较以检测两个图中对应元素是否匹配,并输出匹配成功的部分。

Yang 等[46]介绍了基于遗传算法的流程挖掘、基于日志分类的挖掘算法和基于执行模式的挖掘算法等流程挖掘算法,并从日志完整性、控制流结构、处理噪声、模型质量控制等方面对它们进行了分析和比较,结果表明针对不同特征的日志,需要选择最佳的挖掘算法,这为本文从事件日志片段中提取过程模型提供了挖掘依据。

结束语 随着社会制度的不断完善,法律法规的不断健全,企业的经营管理流程越来越需要满足更多方面法律法规的要求,合规性检测要求得到了比以往任何时候都更高的关注。本文通过理论分析并加以实验验证了所提出的基于图同构的合规性检测框架在进行合规性检测时的可行性。该方法不仅适用于处理已知语义信息的流程模型合规性检测问题,同时还适用于处理语义信息不明的流程模型合规性检测问题。已知语义信息的流程模型在应用图同构技术进行合规性检测时,节点或边的语义信息可以有效减少检测过程探索的状态空间数量;而即使对于无任何语义信息的模型,基于图同构的合规性检测框架也可以通过节点与边的连接结构进行模型匹配,以完成合规性检测。子图同构合规性检测方法为合规性检测技术提供了一种解决方案。在分析事件日志合规性过程中,本文构造扩展可达图以支持可达图对更多维度的规则表达,通过检测 $GNP|=GNR$ 是否成立以判断基于含时间戳的事件日志跟踪的数据流通过程是否是符合基于规则的 Petri 网描述的规则要求。利用图的点与边连接结构是否相同可以判定事件语义无关的合规性检测的功能性属性的合规性问题,通过弧含义相符判定可以进一步丰富合规性检测的非功能性属性检测。

由于子图同构合规性检测方法关注的是图的结构及图中点与边的连接关系,因此即使在不确定检测模型语义信息的情况下,也可以通过对比图的结构以及图中点与边的连接关系,将事件日志中的事件与过程模型中的活动进行对比,从而找到记录在日志中的实际情况和建模行为之间的共性和差异,实现合规性检测。

鉴于合规性检测的重要性,在未来的工作中,将进一步研究提升算法效率的方法,进一步探寻降低算法的时间复杂度的方法,以达到继续提升合规性检测效率的目标。

参 考 文 献

[1] SOX. Sarbanes-Oxley Act[EB/OL]https://www.soxlaw.com/.

[2] Basel Committee on Banking Supervision Basel III [M]. China Financial Publishing House,2014.

[3] General Data Protection Regulation[EB/OL]. https://gdpr.eu/.

[4] Network Security Law of the People's Republic of China[EB/OL]. [2017-02-20]. http://www.npc.gov.cn/wxzl/gongbao/2017-02/20/content_2007531.htm.

[5] Civil Code of the People's Republic of China[EB/OL]. [2020-06-02]. http://www.npc.gov.cn/npc/c30834/202006/75ba6483b8344591abd07917e1d25cc8.shtml.

[6] Data Security Law of the People's Republic of China[M]. Law Press,2021.

[7] Personal Information Protection Law of the People's Republic of China[EB/OL]. [2021-08-20]. http://www.npc.gov.cn/npc/c30834/202108/a8c4e3672c74491a80b53a172bb753fe.shtml.

[8] VAN DER AALST W. Process Mining:Discovery,Conformance and Enhancement of Business Processes[M]. Berlin Heidelberg: Springer-Verlag,2011.

[9] VAN DER AALST W. Process mining[M]. Tsinghua University Press,2014.

[10] RAMEZANI T E,FAHLAND D,AALST V D W M. Where did I misbehave? Diagnostic information in compliance checking[J]. Lecture Notes in Computer Science,2012,7481:262-278.

[11] GHEZZI C,GUINEA S. Run-time monitoring in service-oriented architectures[M]. Heidelberg: Springer Berlin, 2007: 237-264.

[12] EL GAMMAL A F S A. Towards a comprehensive framework for business process compliance[D]. Tilburg University,School of Economics and Management,2012.

[13] WU Z H. Introduction to Petri Nets[M]. China Machine Press, 2006.

[14] PETRI C A. Kommunikation mit Automaten(Communication with Automata)[D]. University of Bonn,1962.

[15] LI H, ZHAO Y L, ZHOU Y L. Time Petri-net models of time ralated system[J]. Journal of Inner Mongolia University(Natural Science Edition), 2000, 31(1): 125-131.
[16] LIU B. Software Verification and Validation[M]. National Defense Industry Press, 2011.
[17] TSAI J J P, JENNHWA Y S, CHANG Y. Timing constraint Petri nets and their application to schedulability analysis of real-time system specifications[J]. IEEE Transactions on Software Engineering, 1995, 21(1): 32-49.
[18] SONG W, DOU W C, LIU X P. Time Constrained Petri Nets and Its Schedulability Analysis and Verification[J]. Journal of Software, 2007(1): 11-21.
[19] LIU X M, LI S X, LI W J, et al. A Time Petri Net with Extended Price Information[J]. Journal of Software, 2007, 18(1): 1-10.
[20] BONHOMME P. A symbolic schedulability technique of real-time systems modeled by P-Time Petri nets[C]// IEEE. 2011: 582-587.
[21] BONHOMME P, BERTHELOT G, AYGALINC P, et al. Verification Technique for Time Petri Nets[C]// 2004 IEEE International Conference on Systems, Man and Cybernetics(IEEE Cat. No. 04CH37583). 2004: 4278-4283.
[22] BONHOMME P, HOVSEPIAN A. Towards a new schedulability technique of real-time systems based on difference of constraints system[C]// IEEE. 2012. 599-604.
[23] ZHOU J T, YE X M. Comparison between reachable graph and reachable tree of Petri nets[J]. Journal of Inner Mongolia University(Natural Science Edition), 2000(1): 117-120.
[24] ZHOU J T, YE X M. A Method for Constructing Reachability Graphs of Petri Nets[J]. Journal of Inner Mongolia University (Natural Science Edition), 1999(3): 127-130.
[25] BOUCHENEB H, BERTHELOT G. Towards a simplified building of time Petri Nets reachability graph[C]// Proceedings of 5th International Workshop on Petri Nets and Performance Models. 1993: 46-47.
[26] ISO/IEC 15909-1-2019, Systems and software engineering-High-level Petri nets-Part 1: Concepts, definitions and graphical notation(Second edition)[S]. ISO/IEC, 2019.
[27] ZHANG Y R, LI H, XING Y, et al. Test case generation combining CPN modeling and on the fly method[J]. Journal of Software, 2017, 28(10): 2564-2582.
[28] CPN Tools(Version 4.0)[EB/OL]. http://cpntools.org/.
[29] ROSEN K H. Discrete Mathematics and Its Applications(the 7th edition of the original book)[M]. China Machine Press, 2015.
[30] CORDELLA L P, FOGGIA P, SANSONE C, et al. A(sub)graph isomorphism algorithm for matching large graphs[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2004, 26(10): 1367-1372.
[31] CARLETTI V, FOGGIA P, SAGGESE A, et al. Challenging the Time Complexity of Exact Subgraph Isomorphism for Huge and Dense Graphs with VF3[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018, 40(4): 804-818.
[32] AALST V D W M. Decomposing Petri nets for process mining: a generic approach[J]. Distributed and Parallel Databases: an International Journal, 2013, 31(4): 471-507.
[33] TANG C J, CHEN P, XIANG Y, et al Introduction to Computational Theory(2nd Edition)[M]. China Machine Press, 2006.
[34] WANG J, ZHAN N J, FENG X Y, et al. Overview of formal methods[J]. Journal of Software, 2019, 30(1): 33-61.
[35] AALST W, BEER H, DONGEN B. Process Mining and Verification of Properties: An Approach Based on Temporal Logic [C]// On the Move to Meaningful Internet Systems 2005: CoopIS, DOA, and ODBASE pt. 1. Lecture Notes in Computer Science. 2005.
[36] MONTALI M, PESIC M, AALST W M P V, et al. Declarative specification and verification of service choreographiess[J]. ACM Transactions on the Web, 2010, 4(1): 1-62.
[37] RAMEZANI T E, FAHLAND D, AALST V D W M. Diagnostic information in compliance checking[R]. BPM Center Report BPM-12-11, BPMcenter. org, 2012.
[38] VAN DER AALST W M P, PESIC M, SONG M. Beyond Process Mining: From the Past to Present and Future[C]// CAiSE 2010: Advanced Information Systems Engineering. 2010: 38-52.
[39] VAN DER AALST W M P, SCHONENBERG M H, SONG M. Time prediction based on process mining[J]. Information Systems, 2011, 36(2): 450-475.
[40] HAN D, TIAN Y H, DU Y Y, et al. Service alignment method based on Petri net reachability graph[J]. Computer Integrated Manufacturing Systems, 2020, 26(6): 1589-1606.
[41] ADRIANSYAH A, VAN DONGEN B F, VAN DER AALST W M P. Cost-Based Conformance Checking using the A* Algorithm[R]. Technical Report, BPM Center Report BPM-11-11, BPMcenter. org, 2011.
[42] WANG Y Q, WEN L J, YAN Z Q. Alignment based conformance checking algorithm for BPMN 2.0 model[J]. Journal of Computer Research and Development, 2017, 54(9): 1920-1930.
[43] VAN DER AALST W, 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.
[44] AGOSTINELLI S, MAGGI F M, MARRELLA A, et al. Achieving GDPR Compliance of BPMN Process Models[M]. Cham: Springer International Publishing, 2019: 10-22.
[45] CAPODIECI A, MAINETTI L. A Structured Approach to GDPR Compliance[M]. Digital Transformation of Collaboration, 2020.
[46] YANG L Q, KANG G S, CAI W G, et al. Research on Business Process Mining Algorithm[J]. Computer Applications and Software, 2016, 33(4): 44-50.

LIU Zhenyu, born in 1987, Ph. D candidates. His main research interests include formal methods and software test.

LI Hua, born in 1964, Ph. D, professor, Ph.D supervisor, is a member of China Computer Federation. Her main research interests include integration of network and cloud computing, big data analysis and evaluation methods, software service computing and testing.