协同业务过程建模与行为验证

赵 莹[1] 潘 华[2] 张云猛[2] 莫 启[3] 代 飞[4]

(云南电力调度控制中心 昆明 650011)[1] (云南云电同方科技有限公司 昆明 650217)[2]

(云南大学软件学院 昆明 650091)[3] (西南林业大学大数据与智能工程学院 昆明 650224)[4]

摘 要 对协同业务过程进行建模和行为验证是确保业务过程正确实施的关键。文中提出了一种协同业务过程的建模和行为验证方法。首先,该方法使用有限状态自动机建模每个参与组织的业务过程,并通过集中式消息缓冲区,将业务过程异步组合为协同业务过程;其次,提出了行为约束的声明式模板,用于定义协同业务过程中的行为约束关系,并通过映射规则,将行为约束关系转换为 LTL(Linear Temporal Logic)公式;最后,提出了行为验证框架,借助进程分析工具 PAT,实现了对协同业务过程行为的自动验证。通过对电力突发公共事件应急处置系统的建模与行为验证,阐述了所提方法的可行性和有效性。

关键词 协同业务过程,业务过程,异步消息通信,行为验证,模型检测

中图法分类号 TP311 **文献标识码** A

Modeling and Behavior Verification for Collaborative Business Processes

ZHAO Ying[1] PAN Hua[2] ZHANG Yun-meng[2] MO Qi[3] DAI Fei[4]

(Yunnan Power Dispatching and Control Center, Kunming 650011, China)[1]

(Yunnan Yundian Tongfang Technology Co., Ltd., Kunming 650217, China)[2]

(School of Software, Yunnan University, Kunming 650091, China)[3]

(School of Big Data and Intelligence Engineering, Southwest Forestry University, Kunming 650224, China)[4]

Abstract Modeling and behavior verification for collaborative business processes is the key to ensure enactment right of business process. This paper proposed an approach to model and verify behavior of collaborative business processes. Firstly, this method uses finite state automaton to model each peer's business process and composes them into the collaborative business process under the asynchronous communication model through the centralized message buffer. Secondly, the declarative template is given for behavior constraint, which is used to define the behavior constraint relationship in collaborative business processes. This behavior constraint specification can be converted to LTL formula by mapping rules. Finally, the framework of behavior verification is proposed to automatically check the behavior of collaborative business processes with the help of PAT (Process Analysis Toolkit). The feasibility and effectiveness of this method were proved through the modeling and behavior verification of emergency response system for public emergency public events.

Keywords Collaborative business process, Business process, Asynchronous message communication, Behavior verification, Model checking

协同业务过程使组织业务过程同其他组织的业务过程进行交互,以形成相对稳定的过程视图,从而满足共同的商业目标[1]。随着全球经济化的发展和企业信息化程度的不断提高,企业的经营模式发生了重大的变化,企业的业务活动已从企业内单目标为导向的独立发展模式发展成为跨企业多目标合作的协同模式[2]。近年来,大量的行业涉及业务协同,如电子商务[3]、供应链[4]以及应急处置系统[5]等。作为一种重要的业务协同使能技术,协同业务过程允许参与组织间共享各种能力(如计算能力、存储能力,甚至是管理能力等),以达到提高计算效率、降低实施成本及实现共赢的目标。由于协同业务过程涉及多个参与组织的业务过程,过程间的交互关系复杂,因此,如何对协同业务过程进行建模和行为验证便成为了业务过程管理领域的热点。

在协同业务过程的建模方面,文献[8-11]主要采用"点对点式"的异步通信模型,将参与组织的业务过程组合为协同业务过程,但未涉及"集中式"的异步通信模型。

在协同业务过程的分析方面,文献[8,10,12]主要从性质(合理性和相容性)角度对协同业务过程进行分析,但普遍缺乏对行为的分析。

针对上述问题,本文提出了一种协同业务过程的建模和

本文受国家自然科学基金 (61462095,61702442),云南省自然科学基金 (2016FB102)资助。

赵 莹(1983—),女,高级工程师,主要研究方向为电力调度自动化与信息化;**潘 华**(1983—),男,高级工程师,主要研究方向为电力调度信息系统;**张云猛**(1987—),男,主要研究方向为电力调度信息系统研究与管理;**莫 启**(1986—),男,博士,讲师,主要研究方向为业务过程和软件工程;**代 飞**(1982—),男,博士,副教授,主要研究方向为业务过程和软件工程,E-mail:59671019@qq.com(通信作者)。

行为验证方法，主要贡献如下：

(1)提出集中式的异步通信模型，将有限状态自动机建模的业务过程异步组合为协同业务过程；

(2)提出行为约束的声明式模板，用于定义协同业务过程应满足的行为约束，并建立了模板和线性时序逻辑公式(Linear Temporal Logic，LTL)[6]间的映射，屏蔽了底层形式化基础的复杂性；

(3)提出了行为验证的框架，使用模型检测技术，借助进程分析工具(Process Analysis Toolkit，PAT)[7]，实现了对协同业务过程行为的自动验证。

本文第 1 节为相关工作；第 2 节讨论协同业务过程的建模；第 3 节提出了行为约束的声明式模板；第 4 节提出行为验证的框架；第 5 节给出了实例分析；最后总结全文。

1 相关工作

在协同业务过程的建模方面，文献[8]采用工作流网 WF-net(workflow net)建模参与组织的业务过程，通过异步库所的增加，在异步通信模型下，提出了 IOWF(inter-organizational workflow)用于建模协同业务过程。文献[9]将 Petri 网和 Pi 演算进行结合，提出了一种协同业务过程建模方法。该方法采用 Petri 网建模参与组织的业务过程，使用 Pi 演算建模参与组织过程间的交互协议，提出了将本地业务过程与交互协议进行融合的映射规则。文献[10]从消息和资源要素两个方面，对工作流网进行扩展，提出了一种资源消息工作流程网 RM_WF_Net(Resource and Message WF-net)，用于建模参与组织的业务过程，并通过库所融合技术，将业务过程组合为协同业务过程。文献[11]提出面向交互的 Petri 网 IOPN(Interaction-Oriented Petri Nets)，用来建模协同业务过程。上述建模工作对协同业务过程的建模具有一定的指导作用和借鉴意义。但是，这些工作主要采用“点对点式”异步通信模型，将导致消息缓冲区众多和无法对通信消息进行统一管理的问题。

在协同业务过程的分析方面，文献[8]采用传统工作流网合理性(Soundness)定义对协同业务过程进行合理性分析，即若每个参与组织的业务过程是合理的，则协同业务过程也是合理的。文献[10]在不考虑消息和资源的前提下，把对协同业务过程的合理性分析转换为对每个参与组织的业务过程进行了合理性分析；在考虑消息和资源的情况下，把对协同业务过程进行的合理性分析转换为对每个参与组织业务过程的合理性分析且协作中产生的消息均被接收。文献[12]为了确保 Web 服务的兼容性，提出了一种基于进程代数的验证方法。首先，该方法将每个 Web 服务的流程转换为 Pi 演算进程表达式；其次，将每个 Web 服务流程对应的 Pi 演算进程表达式并发组合为复合进程；最后，通过展开律自动验证 Web 服务间的兼容性。但是，这些工作只考虑了对协同业务过程进行性质分析，未涉及行为分析。

2 协同业务过程建模

业务过程作为构建协同业务过程的基本单元，用来建模参与组织在过程交互和协作中对外暴露的通信信息。本质上，该业务流程刻画了参与组织内部任务间的执行次序关系。由于自动机具有直观的图形表示，且适合描述业务过程的状态转换，因此本文采用有限状态自动机来建模业务过程。

定义 1(业务过程) 业务过程是一个六元组 $BP=(S, s_s, s_o, \Delta, T, M)$，其中：

(1)S 是有限状态集合；

(2)$s_s, s_o \in S$ 是两个特殊状态，分别表示业务过程的开始状态和终止状态；

(3)$\Delta \subseteq (S \times (\{!,?\} \times T \cup \{\varepsilon\}) \times S)$ 是迁移关系集合；

(4)T 是有限任务集合；

(5)M 是有限消息集合。

引入了两个辅助函数：$send(t)$ 和 $receive(t)$，分别表示任务 t 的发送消息集合和接收消息集合，$send(t): T \rightarrow M^*$；$receive(t): T \rightarrow M^*$。

在协同业务过程中，每个参与组织的业务过程需要跨越组织的边界，同其他组织的业务过程进行通信，以实现共同的商业目标。针对“点对点式”异步通信模型(见图 1)存在的问题：消息缓冲区众多和无法对通信消息进行统一管理，本文提出了“集中式”异步通信模型，如图 2 所示。

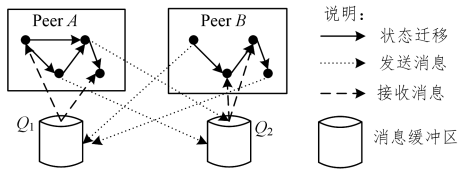


图 1 “点对点式”异步通信模型

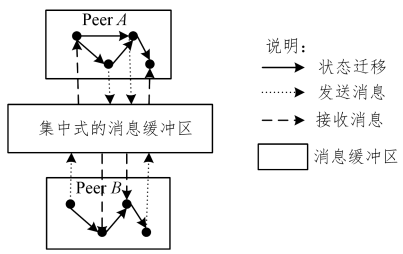


图 2 “集中式”异步通信模型

两种异步通信模型的主要区别在于：1)“点对点式”异步通信模型要求每个参与组织都有一个消息缓冲区，如图 1 中的 Q_1 和 Q_2，而“集中式”异步通信模型只需要一个公共的集中式的消息缓冲区；2)在“点对点式”异步通信模型中，通信的消息分布在不同的消息缓冲区中，不利于统一管理，而在“集中式”异步通信模型中，通信的消息集中在一个消息缓冲区中，利于统一管理；3)从技术实现的角度来看，点对点式异步通信模型要求每个参与组织都要实现消息缓冲区，而“集中式”异步通信模型可以使用各种主流云计算平台提供的企业服务总线以实现集中式的消息缓冲区。

基于“集中式”异步通信模型，下面给出协同业务过程的定义。

定义 2(协同业务过程) 给定业务过程的集合 $BP_s=\{BP_1, BP_2, \cdots, BP_n\}$，每个业务过程表示为 $BP_i=(S_i, s_{si}, s_{oi}, \Delta_i, T_i, M_i)$，$CBP=(BP=, C, c_s, c_o, M_c, T_c, AC)$ 为 BP_s 上的协同业务过程，其中：

(1)$C \subseteq S_1 \times S_2 \times \cdots \times S_n \times Q$ 为状态集合，其中 $Q \subseteq (M_c)^*$；

(2)$c_s, c_o \in C$ 是两个特殊状态，分别表示协同业务过程的开始状态和终止状态；

(3)$M_c = M_1 \cup M_2 \cup \cdots \cup M_n$ 为消息集合；

(4)$T_c = T_1 \cup T_2 \cup \cdots \cup T_n$ 为任务集合；

(5)$AC\subseteq C\times(\{!,?\}\times T_c\cup\{\varepsilon\})\times C$ 为迁移关系,对于所有的 $c=(s_1,s_2,\cdots,s_n,Q)$,$c'=(s_1',s_2',\cdots,s_n',Q')\in T$,有下述 3 种形式中的一种。

1)发送消息动作:$c\xrightarrow{!t}c'$ 有 $\exists i,j\in[1,\cdots,n]$,$\exists t\in T_c$:$t\in T_i\wedge T_i\cap T_j=\{t\}$ 满足:

①$s_i\xrightarrow{!t}s_i'\in\Delta_i$;

②$Q'=Q\cup send(t)$;

③$\forall k\in[1,\cdots,n]:k\neq i\Rightarrow s_k'=s_k$。

2)接收消息动作:$c\xrightarrow{?t}c'$ 有 $\exists i,j\in[1,\cdots,n]$,$\exists t\in T_c$:$t\in T_i\wedge T_i\cap T_j=\{t\}$ 满足:

①$s_i\xrightarrow{?t}s_i'\in\Delta_i$;

②$Q'=Q-receive(t)$;

③$\forall k\in[1,\cdots,n]:k\neq i\Rightarrow s_k'=s_k$。

3)内部动作:$c\xrightarrow{\varepsilon}c'$ 满足:

①有 $\exists i\in[1,\cdots,n]$ 且 $s_i\xrightarrow{\varepsilon}s_i'\in\Delta_i$;

②$\forall k\in[1,\cdots,n]:Q'=Q$;

③$\forall k\in[1,\cdots,n]:k\neq i\Rightarrow s_k'=s_k$。

3　行为约束的声明式模板

为了兼顾易用性和形式验证的需要,声明式业务规约描述语言(如 ConDec[13],SCIFF[14] 和 DCR Graphs[15] 等)不断被提出。这些声明式业务规约描述语言通过建立声明式模板与形式模型间的映射规则,使得业务分析人员无需了解底层形式化基础,只需采用声明式模板就能准确描述业务需求。对业务分析人员而言,声明式业务规约描述语言屏蔽了底层形式化基础的复杂性,提高了定义业务规约的简易性。

协同业务过程的行为表现为业务过程间的消息通信,而业务过程间的消息通信又由发送消息动作和接收消息动作所刻画。因此,对协同业务过程进行行为验证的关键在于定义消息动作间的时序约束。

借鉴文献[16]中提出的控制流合规性模式思想,本文提出了行为约束的声明式模板,如表 1 和表 2 所列,用于定义消息动作间的二元约束关系和多元约束关系。二元关系模板定义了两个消息动作间需满足的次序关系。多元关系模板定义了多个消息动作间需满足的次序关系。

在表 1 和表 2 中,a 和 b 表示消息动作;A 和 B 表示消息动作的集合。此外,可以直观看到,表 1 和表 2 所列声明式模板均被转换为 LTL 公式。关于 LTL 的语法和语义请参考文献[6]。这为将来我们使用模型检测工具对行为进行验证奠定了基础。

需要指出的是,尽管表 1 和表 2 列出了实际应用中常见的发送消息动作间的二元约束关系和多元约束关系,但我们还无法给出数学证明说明这两种约束关系的完备性。这也是我们后续工作的关注重点。

表 1　二元关系的声明式模板

模板	模板描述	映射 LTL 公式
$CoExist(a,b)$	a 产生,则 b 产生,反之亦然	$\Diamond a\rightarrow\Diamond b$
$Resp(a,b)$	a 产生,则 b 必定产生	$\Box(a\rightarrow\Diamond b)$
$ChainPesp(a,b)$	a 产生,则 b 必定下一个时刻产生	$\Box(a\rightarrow\bigcirc b)$
$Prior(a,b)$	a 在发送 b 之前产生	$\Diamond b\rightarrow(\neg b\cup a)$
$ChainPrior(a,b)$	a 在 b 之前一个时刻产生	$Prior(a,b)\wedge\Box(\bigcirc b\rightarrow a)$
$Alter(a,b)$	两个 a 发生之间至少存在一个 b 产生	$Resp(a,b)\wedge\Box(a\rightarrow\bigcirc Prior(a,b))$
$NotCoExist(a,b)$	a 或者 b 发生,但不能同时产生	$\Diamond a\rightarrow\neg\Diamond b\vee\Diamond b\rightarrow\neg\Diamond a$
$NotResp(a,b)$	a 产生,则 b 必定不能产生	$\Box(a\rightarrow\neg\Diamond b)$
$NotChain(a,b)$	a 产生,则 b 必定不在下一个时刻产生	$\Box(a\rightarrow\neg\bigcirc b)$
$NotAlter(a,b)$	两个 a 发生之间不存在 b 产生	$\Box(a\rightarrow\bigcirc(\neg b\cup a))$

表 2　多元关系的声明式模板

模板	模板描述	映射 LTL 公式
$Option(A)$	A 中的一个或多个发送消息动作产生	$\Diamond(a_1\vee\cdots\vee a_n)$
$MutiOpResp(a,B)$	若 a 产生,则 B 中至少一个发送消息动作产生	$\Box(a\rightarrow\Diamond(b_1\vee\cdots\vee b_n))$
$MutiResp(a,B)$	若 a 产生,则 B 中所有发送消息动作产生	$\Box(a\rightarrow\Diamond(b_1\wedge\cdots\wedge b_n))$
$MutiOpPrior(A,b)$	若 b 产生,则 A 中至少一个发送消息动作在此之前产生	$\Diamond b\rightarrow(\neg b\cup(a_1\vee\cdots\vee a_n))$
$MutiPrior(A,b)$	A 中所有发送消息动作在 b 之前产生	$\Diamond b\rightarrow(\neg b\cup(a_1\wedge\cdots\wedge a_n))$
$NotMuOpRp(a,B)$	若 a 产生,则 B 中所有发送消息动作不能产生	$\Box(a\rightarrow\Box\neg b_1\wedge\cdots\wedge\Box\neg b_n))$
$NotMuRp(a,B)$	若 a 产生,则不是所有 B 中发送消息动作不能产生	$\Box(a\rightarrow\Box(\neg b_1\vee\cdots\vee\neg b_n))$
$NotMuOpPro(A,b)$	若 b 产生,则 A 中所有发送消息动作在此之前不能产生	$\Box((a_1\vee\cdots\vee a_n))\rightarrow\neg\Diamond b)$
$NotMuPro(a,b)$	若 b 产生,则不是所有 A 中发送消息动作在此之前不能产生	$\Box((a_1\wedge\cdots\wedge a_n))\rightarrow\neg\Diamond b)$

4　行为验证框架

为了实现对协同业务过程行为的自动验证,本文基于模型检测思想[17],给出了行为验证的框架,如图 3 所示。具体而言,包含 3 个步骤:

第 1 步　将协同业务过程转换为 CSP 进程[18]。首先,将协同业务过程中的每个业务过程转换为对应的 CSP 进程,具体参见算法 1 的步骤 5－步骤 32;其次,通过提出的消息数值化(参见定义 3),将每个业务过程对应的 CSP 进程组合为并发 CSP 进程,具体参见算法 1 的步骤 20、步骤 28 和步骤 33。

第 2 步　使用行为约束的声明式模型定义协同业务过程需要满足的行为约束规约,根据表 1 和表 2,将该规约转换为对应的 LTL 公式。

第 3 步　将并发 CSP 进程和 LTL 公式作为模型检测器 PAT 的输入,借助工具,使用模型检测技术,检测并发 CSP 进程是否满足 LTL 公式。若不满足,PAT 将给出反例信息,建

模者可根据反例信息进一步对协同业务过程进行修改。

由上述步骤可知,第一步是实现行为验证的关键,重点需要解决两个问题:1)如何将每个参与组织的业务过程转换为 CSP 进程;2)在异步通信环境下,如何将多个 CSP 进程组合为并发 CSP 进程。

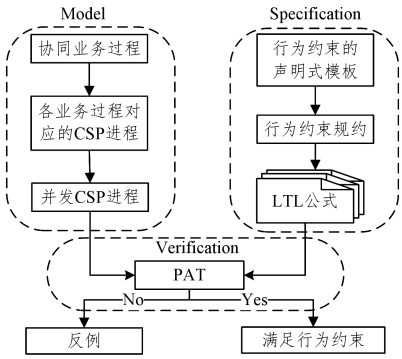


图 3 行为验证框架

本质上,将每个参与组织的业务过程转换为 CSP 进程是把一个有限状态自动机转换为 CSP 进程,其基本思想是:1)找出自动机中具有唯一后继状态的状态,将引起该状态发生迁移的唯一任务转换为 CSP 进程中的前缀动作;2)找出自动机中具有多个后继状态的状态,将引起该状态发生迁移的多个任务分别转换为选择动作。

在异步通信环境下,将多个 CSP 进程组合为并发 CSP 进程的基本思想是:将协同业务过程中发送和接收的消息集合矢量化视为一个多维矢量,若某条消息在协作中被发送,则该消息对应的维度上的数值加 1;若某条消息在协作中被接收,则该消息对应的维度上的数值减 1。

定义 3(消息数值化) 设 $CBP=(BPS,C,c_s,c_o,M_c,T_c,AC)$为协同业务过程,$\vec{V}$为一个 n 维向量,$n=|M_c|$,$fa:M_c\rightarrow N$ 为消息数值化函数,其中 N 为整数,有:

(1)若 $c\xrightarrow{!t}c'\in AC$,则 $send(t)$中每个消息在$\vec{V}$中对应的维度上的数值加 1;

(2)若 $c\xrightarrow{?t}c'\in AC$,则 $receive(t)$中每个消息在$\vec{V}$中对应的维度上的数值减 1。

基于上述思想,本文给出的算法 1 用于将协同业务过程转换为并发 CSP 进程。其基本思想是:针对协同业务过程中的每个业务过程 BP_i,采用广度优先搜索策略生成其对应的 CSP 进程表达式。在 CSP 进程表达式中,每个名字对应业务过程中的一个任务。任务执行前将需要接收的消息集转换为该名字前的关联约束条件;任务执行后将发送的消息集转换为该名字后的关联约束条件。最后,通过多维矢量和消息数值化,将每个业务过程转换生成的 CSP 进程表达式并发组合为协同业务过程对应的并发 CSP 进程。

若协同业务过程中含有 n 个业务过程,业务过程的平均深度和平均状态迁移数分别为 m 和 k,则算法 1 的时间复杂度为 $O(n*m*k)$。

算法 1 在协同业务过程中产生 CSP 进程

Input:协同业务过程 CBP=(BPS,C,c_s,c_o,M_c,T_c,AC)

Output:CSP 进程

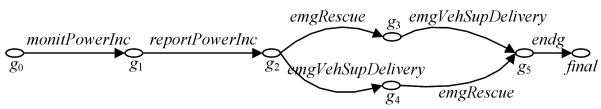
1. get M_c of CBP;
2. for each msg in M_c do//将 M_c 映射为一个多维矢量,且每个维度上的初值为 0
3. buffer.add(msg=0);
4. end for
5. for each BP_i in BPS do//遍历每个参与组织的业务过程
6. put s_s of BP_i into queue Q;
7. put s_s of BP_i into visited queue VQ;
8. while Q.size>0 then
9. elem=Q.poll;
10. for each r in BP_i.Δdo //获取从 elem 出发的状态迁移
11. if elem 为迁移 r 中的初始状态 then
12. S.add(r);
13. end if
14. end for
15. if S.size==0 then
16. add(elem,ε,elem) to buffer;
17. end if;
18. if S.size==1 then //构建 CSP 进程 $P(s_s)$的前缀动作
19. r=S.get(0);
20. 得到 r 中任务 a 的 send(a)和 receive(a);
generate $P(s_s)=[msg_{1r}>0\ \&\&\cdots\&\&\ msg_{mr}>0]$// send(a)映射为守卫
a
{ $msg_{1r}--\&\&\cdots\&\&\ msg_{mr}--\&\&$ // receive(a)对应的矢量维度上的数值减 1
$msg_{1s}++\ \&\&\cdots\&\&\ msg_{ns}++$}→$s_e$// send(a)对应的矢量维度上的数值加 1
add $P(s_s)$to buffer;
21. if VQ does not contains s_s then
22. add s_s to Q;
23. add s_s to VQ;
24. end if
25. end if
26. if S.size> 1 then //构建 CSP 进程 $P(s_s)$的选择动作
27. $S=\{r_1=(s_s,a_1,s_{e1}),\cdots,r_k=(s_s,a_k,s_{ek})\}$;
28. generate $P(s_s)=[msg_{11r}>0\ \&\&\cdots\&\&\ msg_{mrl}]\ a\{msg_{1sl}>0\ \&\&\cdots\&\&\ msg_{nsl}\}\rightarrow s_{e1}[]\cdots[][msg_{1rk}>0\ \&\&\cdots\&\&\ msg_{mrk}]\ a\{msg_{1sk}>0\ \&\&\cdots\&\&\ msg_{nsk}\}\rightarrow s_{ek}$;
29. add $P(s_s)$ to buffer;
30. end if
31. end while
32. end for
33. generate Pro=$i_0\parallel\cdots\parallel\ i_n$;//构建并发 CSP 进程 Pro
34. add Pro to buffer;
35. return buffer.

5 实例分析

应急处置系统是一类典型的业务协同系统。因此,本文通过对某地区电力突发公共事件应急处置系统进行交互行为正确性验证来阐述本文方法的有效性。该应急处置系统包含的参与组织有:电力部门、应急指挥部办公室、卫生部门、公安部门以及宣传部门。电网公司实时对电力突发事件进行监控(monitPowerInc),当发现突发事件时及时向应急指挥部办公室上报突发事件情况(reportPowerInc);应急指挥部办公室在

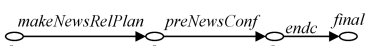
接收到突发事件(recPowerInc)后组织专家制定并向电力部门、卫生部门、公安部门以及宣传部门下达应急处置方案(madeRelPlan);电力部门在接收应急处置方案后同时开展抢险救援(emgRescue)和应急车辆及应急物资派送(emgVehSupDelivery);卫生部门在接收应急处置方案后立即派医疗力量赶赴现场(mdeRushSite),接着同时开展现场医疗救治(treatOnSite)和伤员转运(wouTransfer)工作,并向宣传部门汇报伤员情况(reportWouCond);公安部门在接收应急处置方案后立即派公安力量赶赴现场(polRushSite);接着同时开展维持现场次序(matSizeOrder)和疏散群众(evaMasses)工作,并向宣传部门汇报现场情况(reportSizeCond);宣传部门在接收应急处置方案后立即制定新闻发布方案(makeNewsRelPlan),并在接收到卫生部门和公安部门报告的相关情况后适时召开新闻发布会(preNewsConf)。

(1)根据上述电力突发公共事件应急处置系统分析,可建模得到电网公司、应急指挥部办公室、卫生部门、公安部门以及宣传部门各自的业务过程,分别如图 4 所示。

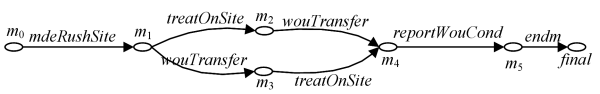


(a)电网公司业务流程

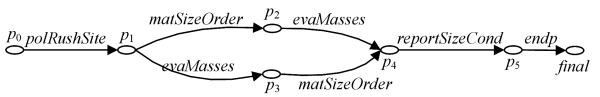
(b)应急指挥办公室业务流程



(c)宣传部业务流程



(d)卫生部业务流程



(e)公安部业务流程

图 4　参与组织的业务过程

(2)以电网公司、应急指挥部办公室、卫生部门、公安部门和宣传部门的业务过程为基础,整个应急处置系统的协同业务过程定义为 $CBP=(BP_S, C, c_s, c_o, M_c, T_c, AC)$,其中:$BP_S=\{BP_{pow}, BP_{emg}, BP_{mde}, BP_{pol}, BP_{pub}\}$,$BP_{pow}$,$BP_{emg}$,$BP_{mde}$,$BP_{pol}$,$BP_{pub}$ 分别为以自动机表示的电网公司、应急指挥部办公室、卫生部门、公安部门和宣传部门的业务过程,如图 4 所示。

(3)根据算法 1 将 *CBP* 转换成并发 CSP 进程,如下所示。

varpowerInc=0;

varmdePlan=0;

varpolPlan=0;

varnewsPlan=0;

varemgResPlan=0;

varemgVehSupPlan=0;

varwouCond=0;

varsizeCond=0;

GridDep()=monitPowerInc→G1();

G1()=reportPowerInc{powerInc++}→G2();

G2()=[emgResPlan>0]emgRescue{emgResPlan--}→

G3()[][emgVehSupPlan>0]emgVehSupDelivery

{emgVehSupPlan--}→G4();

G3()=[emgVehSupPlan>0]emgVehSupDelivery

{emgVehSupPlan--}→G5();

G4()=[emgResPlan>0]emgRescue

{emgResPlan--}→G5();

G5()=endg→Skip;

CmdDep()=[powerInc>0]recPowerInc

{powerInc--}→H1();

H1()=madeRelPlan{mdePlan++;polPlan++;

newsPlan++;emgResPlan++;

emgVehSupPlan++}→H2();

H2()=endh→Skip;

MedDep()=[mdePlan>0]mdeRushSite

{mdePlan--}→M1();

M1()=treatOnSite→M2() [] wouTransfer→M3();

M2()=wouTransfer→M4();

M3()=treatOnSite→M4();

M4()=reportWouCond{wouCond++}→M5();

M5()=endm→Skip;

PolDep()=[polPlan>0]polRushSite{polPlan--}→P1();

P1()=matSizeOrder→P2() [] evaMasses→P3();

P2()=evaMasses→P4();

P3()=matSizeOrder→P4();

P4()=reportSizeCond{sizeCond++}→P5();

P5()=endp→Skip;

CapDep()=[newsPlan>0]makeNewsRelPlan

{newsPlan--}→C1();

C1()=[wouCond>0 &&sizeCond>0]preNewsConf

{wouCond--;sizeCond--}→C2();

C2()=endc→Skip;

ERS()=GridDep() ‖ CmdDep()‖MedDep()‖PolDep() ‖ CapDep();

其中,ERS()定义了此应急处置系统对应的 CSP 进程表达式。

(4)设业务设计人员采用交互行为声明式模板定义的行为约束规约如下:

1)Prior(powerInc,mdePlan),表示应急指挥部办公室只有在收到突发事件后才能向医疗部门下达应急处置方案,映射生成的 LTL 公式为◇mdePlan→(¬mdePlanupowerInc)。

2) MutiResp(powerInc,{mdePlan, polPlan, newsPlan, emgResPlan,emgVehSupPlan}),表示应急指挥部办公室在收到突发事件后同时向电力部门、卫生部门、公安部门以及宣传部门下达应急处置方案,映射的 LTL 公式为□(powerInc→◇(mdePlan ∧ polPlan ∧ newsPlan ∧ emgResPlan ∧ emgVehSupPlan))。

(5)将上述进程输入到 PAT 中,保存为 ERS.csp 文件,在检查进程文件没有语法错误后,对 ERS()的行为进行验证,验证结果如图 5 所示。

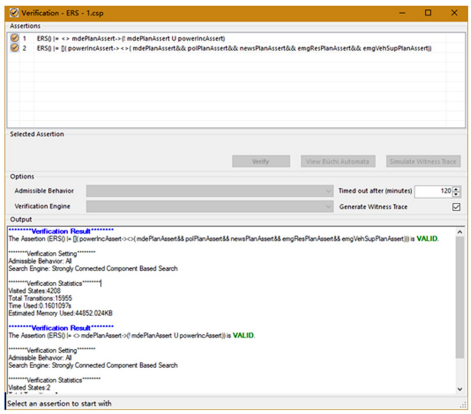


图 5 结果界面

图 5 给出了两条行为约束的验证结果均为“Valid”。由此可以确定该电力突发公共事件应急处置系统中的消息执行与预期一致，说明该应急处置系统的设计是合理的。

结束语 针对协同业务过程的建模和分析，首先，本文使用有线状态自动机建模参与组织的业务过程，通过“集中式”异步通信模型，将业务过程组合为协同业务过程；其次，提出行为约束的声明式模板，用于定义协同业务过程需要满足的行为约束规约；最后，使用模型检测技术，借助 PAT 工具，实现了对协同业务过程行为的自动验证。

本文的工作关注协同业务过程的建模和行为验证，未来的工作将重点考虑行为验证面临的状态爆炸问题。

参 考 文 献

[1] 代飞，莫启，林雷蕾，等. 结合 Petri 网和 Pi 演算的协同业务过程建模[J]. 计算机科学与探索，2015，9(6)：692-706.

[2] 卢亚辉，明仲，张力. 业务过程协同模式的研究[J]. 计算机集成制造系统，2011，17(8)：1570-1579.

[3] YU W Y，YAN C G，DING Z J，et al. Modeling and verification of online shopping business processes by considering malicious behavior patterns[J]. IEEE Transactions on Automation Science and Engineering，2016，13(2)：647-662.

[4] SHAHRIARI K，HESSAMI A G，JADIDI A，et al. An approach toward a conceptual collaborative framework based on a case study in a wood supply chain[J]. IEEE Systems Journal，2015，9(4)：1-10.

[5] 曾庆田，鲁法明，刘聪，等. 基于 Petri 网的跨组织应急联动处置系统建模与分析[J]. 计算机学报，2013，36(11)：2290-2302.

[6] KESTEN Y，PNUELI A，RAVIV L O. Algorithmic verification of linear temporal logic specifications[J]. Lecture Notes in Computer Science，1999，1443(1443)：1-16.

[7] CS Department NUS. PAT：Process Analysis Toolkit [EB/OL]. [2013-09-13]. http://www.patroot.com.

[8] AALST W. Modeling and analyzing interorganizational workflows[C]// Proc of the 1st IntConf on Application of Concurrency to System Design. Los Alamitos，CA：IEEE Computer Society，1998：262-272.

[9] ZHANG L，LU Y，XU F. Unified modelling and analysis of collaboration business process based on Petri nets and Pi calculus [J]. IET Software，2010，4(5)：303-317.

[10] ZENG Q T，LU F M，LIU C，et al. Modeling and verification for cross-department collaborative business processes using extended Petri nets[J]. IEEE Trans on Systems，Man，and Cybernetics：Systems，2015，45(2)：349-362.

[11] 葛季栋，胡海洋，周宇，等. 一种基于不变量的工作流协同模型分解方法[J]. 计算机学报，2012，35(10)：2169-2181.

[12] 邓水光，李莹，吴健，等. Web 服务行为兼容性的判定与计算[J]. 软件学报，2007，18(12)：3001-3014.

[13] AALST W，PESIC M，SCHONENBERG H. Declarative workflows：Balancing between flexibility and support[J]. Computer Science-Research and Development，2009，23(2)：99-113.

[14] MONTALI M. Specification and verification of declarative open interaction models-A logic-based approach[J]. Springer Science & Business Media，2010，56(1)：47-76.

[15] HIDEBRANDT T，MUKKAMALA R. Declarative event-based workflow as distributed dynamic condition response graphs[C]// Electronic Proceedings in Theoretical Computer Science (EPTCS) 69：Proc of PLACES 2010. Sydney，Australia：EPTCS，2011：59-73.

[16] AWAD A，WEIDLICH M，WESKE M. Visually specifying compliance rules and explaining their violations for business process [J]. Journal of Visual Languages & Computing，2011，22 (1)：30-55.

[17] BAIER C，KATOEN J P. Principles of model checking[M]. ambridge：MIT Press，2008.

[18] SUN J，LIU Y，DONG J S. Model Checking CSP Revisited：Introducing a Process Analysis Toolkit[M]// Leveraging Applications of Formal Methods，Verification and Validation. Springer Berlin Heidelberg，2008：307-322.

(上接第 583 页)

[4] 何蒲，于戈，张岩峰，等. 区块链技术与应用前瞻综述[J]. 计算机科学，2017，44(4)：6.

[5] 许涛. 区块链技术在教育教学中的应用与挑战[J]. 现代教育技术，2017，27(1)：110-111.

[6] 安瑞，何德彪，张韵茹，等. 基于区块链技术的防伪系统的设计与实现[J]. 密码学报，2017，4(2)：199-208.

[7] 田海博，何杰杰，付利青. 基于公开区块链的隐私保护公平合同签署协议[J]. 密码学报，2017，4(2)：187-198.

[8] 夏新岳. 基于区块链的股权资产购买和转赠设计与实现[D]. 内蒙古：内蒙古大学，2016：29-36.

[9] 黄洁华，高灵超，许玉壮，等. 众筹区块链上的智能合约设计[J]. 信息安全研究，2017，3(3)：211-219.

[10] 张波. 国外区块链技术的运用情况及相关启示[J]. 金融科技时，2016(5)：35.

[11] 黄征，李祥学，来学嘉，等. 区块链技术及其应用[J]. 信息安全研究，2017，3(3)：237-245.

[12] 冯超政，蒋溢，何军，等. 基于冷热数据的 MongoDB 自动分片机制[J]. 计算机工程，2017，43(3)：7-10.

[13] 王亚玲，杨超，章名尚. 数据库系统应用分片中间件[J]. 计算机系统应用，2015，24(10)：76-78.

[14] 吴黎兵，党平，聂雷，等. 一种可分片预留接纳控制算法研究[J]. 计算机研究与发展，2014，51(6)：1201-1204.

[15] 蔡维德，郁莲，玉荣，等. 基于区块链的应用系统开发方法研究[J]. 软件学报，2017，28(6)：1474-1487.