

数据和时间感知的 Web 服务组合模型检测

狄浩军¹ 戎 玫² 张广泉^{1,3} 朱稷涵¹

(苏州大学计算机科学与技术学院 苏州 215006)¹ (暨南大学深圳旅游学院 深圳 518053)²

(中国科学院软件研究所计算机科学国家重点实验室 北京 100080)³

摘 要 为了验证 Web 服务组合中的数据属性和时间属性,提出了一种基于数据和时间感知的服务模型 DTSM 的 Web 服务组合模型检测方法。首先将 BPEL 描述的 Web 服务组合转换为含有数据流信息和时间信息的形式化模型,然后将该模型映射至 UPPAAL 规约,最后应用模型检测工具 UPPAAL 实现了 Web 服务组合的正确性验证。

关键词 Web 服务组合,数据属性,时间属性,BPEL,模型检测

中图法分类号 TP311

文献标识码 A

Model Checking of Data and Time Aware Web Service Composition

DI Hao-jun¹ RONG Mei² ZHANG Guang-quan^{1,3} ZHU Ji-han¹

(School of Computer Science and Technology, Soochow University, Suzhou 215006, China)¹

(Shenzhen Tourism College, Jinan University, Shenzhen 518053, China)²

(State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences, Beijing 100080, China)³

Abstract To validate the data properties and time properties of Web service composition, we presented a model checking method based on data and time aware service model (DTSM). In this approach, first we translated Web service composition described by BPEL to formal model which contains data information and time information, and then translated this model to UPPAAL specification, at last the correctness of Web service composition was verified through a model checking tool—UPPAAL.

Keywords Web service composition, Data property, Time property, BPEL, Model checking

1 引言

近年来,随着互联网的普及和电子商务技术的发展,面向服务的体系结构(Service-oriented Architecture, SOA)正逐步成为未来软件发展的趋势之一。作为实现 SOA 的主流方式,Web 服务技术也得到了快速的发展与应用。Web 服务是一个基于网络的自描述、自包含的软件模块,具有松耦合、独立性强等特点。由于单个 Web 服务的功能有限,通过服务组合的方式来满足用户需求已成为必然趋势。如何保证服务组合的质量问题,也已成为学术界和工业界关注的热点。形式化验证方法是保证软件正确性的重要方法之一,目前常见的形式化验证方法主要分为两类:演绎验证(deductive verification)和模型检测(model checking)。其中,模型检测^[1]是一种自动的、基于模型的性质验证处理方法。该方法的基本思想是用状态迁移系统 S 表示系统的行为,用时序逻辑公式 F 描述系统的性质,这样“系统是否满足所期望的性质”就转化为数学问题“状态迁移系统 S 是否是公式 F 的一个模型”。

由于模型检测技术具有自动化程度高、验证速度快等优点,已成为 Web 服务组合静态验证阶段的关键技术。Web 服务组合的静态验证过程如下:首先,将 Web 服务组合规约(如 BPEL、WS-CDL 等)映射到形式化模型(如进程代数、Petri 网、有限自动机等),并转化为某种形式化规约作为模型检测工具(如 UPPAAL、Spin、NuSMV 等)的输入;其次,将 Web 服务组合待验证性质表示为形式化规约语言,将其也作为模型检测工具的输入;最后,运行模型检测工具,如果输出结果为“yes”,说明系统满足所给性质;输出结果为“no”,说明系统不满足所给性质,并给出导致失败的系统行为轨迹。

2 相关工作

目前采用模型检测技术验证 Web 服务组合属性的研究工作已有很多。文献[2]提出了一种能够分析 BPEL 描述的 Web 服务组合交互过程的技术方法,并使用模型检测工具 Spin 对 Web 服务组合的行为属性进行了分析验证。文献[3]将 WS-CDL 描述的带有时间约束的 Web 服务组合自动转换

到稿日期:2010-12-12 返修日期:2011-03-21 本文受国家自然科学基金(60973149),江苏省自然科学基金(BK2011281),中国科学院计算机科学国家重点实验室开放课题(SYSKF0908),江苏省高校自然科学研究项目(08KJB520010),苏州大学国家级大学生创新性实验计划项目(101028524)资助。

狄浩军(1987—),男,硕士生,主要研究方向为软件服务与形式化方法, E-mail: dihaojun@yahoo.com.cn; 戎玫 博士,副教授,主要研究方向为软件工程与网络分布计算; 张广泉 博士,教授,CCF 高级会员,主要研究方向为服务计算、软件体系结构与形式化方法、网络与分布式计算等; 朱稷涵 男,本科生。

成时间自动机,并使用模型检测工具 UPPAAL 来模拟和分析系统的行为。文献[4]将服务组合描述语言 WS-CDL 转换为模型检测工具 Spin 的输入语言 Promela,然后对系统的死锁、安全性和活性进行了验证。文献[5]采用模型检测技术研究了 Web 服务编排的兼容性问题,同时考虑了服务组合的时间属性。文献[6]提出了一种扩展着色 Petri 网的模型检测方法,将扩展后的 CPN 模型检测工具应用在 Web 服务组合的验证问题中,该方法不仅可以验证 Web 服务组合是否存在逻辑错误,还能告知用户发生错误的原因。文献[7]研究了智能体认知逻辑的模型检测问题,将 Web 服务组合建模为多智能体系统,并采用模型检测工具 MCTK 验证了系统的时态认知逻辑规范。

我们在这方面也做过类似工作。文献[8]从高层抽象层次研究了 Web 服务组合的正确性及其形式化验证方法,采用软件体系结构描述语言 XYZ/ADL 描述服务组合,并用模型检测工具 UPPAAL 实现了 Web 服务组合正确性的自动化验证。

然而在这些研究中,绝大部分都只是考虑对服务组合中行为属性(如活性、安全性等)的分析验证,很少考虑服务交互中的数据流信息,缺乏对服务组合中数据属性的分析验证。另外,在时间属性的验证方面也不是很深入、全面。基于此,本文对有限自动机进行扩展,提出了一种基于数据和时间感知的服务模型(Data and Time Aware Service Model,DTSM)的 Web 服务组合模型检测方法,并分别对服务组合中的行为属性、数据属性和时间属性进行了分析验证。

3 数据和时间感知的服务模型 DTSM

本节介绍数据和时间感知的服务模型 DTSM,该模型能刻画服务交互中的数据流信息和时间信息。在介绍该模型之前,先引入水果销售服务 FSS(Fruit Sales Service)实例作为本文讨论的基础。

如图 1 所示,水果销售服务 FSS 由库存服务 InventoryService 和订单服务 BillingService 采用编制的方式组合而成。在该服务场景中,购买者 Buyer 要想购买水果,可以向 FSS 提交查询请求消息 searchfruits, FSS 根据查询信息会调用 InventoryService 向购买者返回包含水果种类、价格、库存量等详细信息的信息 fruitdetails,或者返回“查询无效”的消息 searchinvalid;购买者若准备购买水果,可以提交一个订购单消息 buyoder, FSS 收到订购单后会调用 BillingService 进行审核。若审核通过,则会发送一个订购单确认消息 oderconfirm,否则返回订购单无效消息 oderinvalid;最后购买者发送支付订单消息 payment 后,将会收到 FSS 验证后的支付确认消息 paymentconfirm,或者支付无效消息 paymentinvalid。另外,在整个交互过程中,购买者 Buyer 向水果销售服务 FSS 发出消息后必须在短时间内收到反馈消息,否则便会取消交易,而 FSS 也会对购买者 Buyer 长时间的无响应做出相应处理。

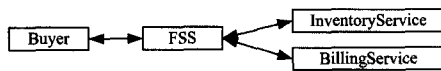


图 1 Web 服务组合实例

以上实例场景展示了对数据流信息和时间信息刻画的需求。在数据流信息方面,整个交互过程中,显然需要满足“购买者 Buyer 提交的订购单消息中包含的水果信息(如水果名

称)必须与支付订单消息中的水果信息保持一致”以及“只有当购买者 Buyer 支付了正确的金额数目时, FSS 才会发送支付确认消息”,以上描述中包含了刻画水果名称、水果数量、单价、总价等隐含的数据流信息的要求。而在时间信息方面,交互过程中需要满足“只有购买者 Buyer 在规定时间内收到水果销售服务 FSS 的反馈后,才会继续进行有效交易”。显然该需求包含了刻画时间信息的要求。

可以看出,这些数据流信息和时间信息对整个服务交互有着重要的影响。正确刻画这些数据流信息和时间信息,是验证与其相关的数据属性和时间属性的必要前提。

为了刻画数据流信息对服务交互的影响,采用消息数据概念来表达数据流信息。我们将服务间传送的消息中所包含的可表达、具有实际意义且能一定程度影响服务交互的数据内容称为消息数据,并将消息抽象为消息数据的集合。

定义 1 设 M 为消息的集合, D 为消息数据变量的集合, V 为消息数据值的集合,定义映射:

$$L: M \times D \rightarrow V$$

对于每一个消息 $m \in M$, $L(m, d_i) \in V$ 表示消息 m 中包含的消息数据 v_i 。另外,我们可以假设 L 唯一地标识了每一个消息,也就是说,不存在消息 $m_1, m_2 \in M$, 有 $L(m_1, d) = L(m_2, d)$, 其中 $m_1 \neq m_2, d \in D$ 。

由于消息中包含的消息数据含有多种数据类型,对于消息数据变量的集合 D 、数据类型的集合 T ,定义映射 $F: D \rightarrow T$,即对于每个消息数据变量 $d \in D$ 都属于数据类型 $F(D)$ 。

在服务建模中,需要刻画在状态迁移过程中对消息数据的约束,而不同类型的消息数据约束实施的操作不同。根据实际需求我们只给出 3 种类型的消息数据约束,即整数型消息数据约束、实数型消息数据约束和字符串型消息数据约束。

定义 2 设 D 为消息数据变量的集合, Φ 为消息数据约束的集合,对于不同类型的消息数据变量:

1) 若 d 为整数型消息数据变量,消息数据约束 φ 的集合记为 $\Phi(D)$, φ 文法定义如下:

$$\varphi ::= d \Theta c \mid d_1 \Theta d_2 \mid \neg \varphi \mid \varphi \wedge \varphi$$

式中,操作符 $\Theta \in \{<, >, \leq, \geq, =, \neq, +, -, \times, \div\}$, d_1, d_2 为整数型消息数据变量, c 为整数常量。

2) 若 d 为实数型消息数据变量,消息数据约束 φ 的集合记为 $\Phi(D)$, φ 文法定义如下:

$$\varphi ::= d \Theta c \mid d_1 \Theta d_2 \mid \neg \varphi \mid \varphi \wedge \varphi$$

式中,操作符 $\Theta \in \{<, >, \leq, \geq, =, \neq, +, -, \times, \div\}$, d_1, d_2 为实数型消息数据变量, c 为实数常量。

3) 若 d 为字符串型消息数据变量,消息数据约束 φ 的集合记为 $\Phi(D)$, φ 文法定义如下:

$$\varphi ::= d = \text{str} \mid d \neq \text{str} \mid d_1 = d_2 \mid d_1 \neq d_2 \mid \neg \varphi \mid \varphi \wedge \varphi$$

式中, d_1, d_2 为字符串型消息数据变量, str 为字符串常量。

为了刻画时间信息对服务交互的影响,我们采用时间自动机中的时钟来表达时间信息。时钟是一个非负实数上的变量,所有的时钟值都以统一的速度流逝。开始时时钟一般都置零,事件发生时,时钟可以被重置为零。一个时钟约束是将时钟值和一个时间常量进行比较,而且时钟约束只能是上述原子形式时钟约束的布尔合取,其定义如下。

定义 3 设 X 为时钟变量的集合, Δ 为时钟变量约束的集合,对于时钟变量 $x \in X$,时钟约束 δ 的集合记为 $\Delta(X)$, δ 文法定义如下:

$$\delta ::= x \leq c \mid x \geq c \mid x < c \mid x > c \mid \neg \delta \mid \delta \wedge \delta$$

式中, c 为非负实数常量。

另外,对于时间自动机中的时钟,一个时钟解释 t 是指为时钟集 X 上的每一个时钟赋予一个实数值,也就是说,它是从 X 到非负实数集 R^+ 的一个映射。 X 的一个时钟解释 t 满足 X 上的一个时钟约束 δ ,当且仅当使用 t 给出的值, δ 为真。

本文对有限自动机进行扩展,采用消息数据来表达数据流信息,采用时间自动机中的时钟来表达时间信息,对 Web 服务进行如下建模。

定义 4 数据和时间感知的服务模型 DTSM 为一个九元组 $\Sigma = (S, s_0, F, M, X, D, I, I', R)$, 其中:

- S 是有穷状态的集合;
- $s_0 \in S$ 是初始状态;
- $F \subseteq S$ 是终止状态的集合;
- M 是消息的集合;
- X 是有穷时钟变量的集合;
- D 是消息数据变量的集合;
- I 是一个映射,它给 S 中的每个状态指定 $\Delta(X)$ 中的一个时钟约束;

• I' 也是一个映射,它给消息中的每个消息数据指定 $\Phi(D)$ 中的一个消息数据约束;

• $R \subseteq S \times M \times \delta(X) \times \varphi(D) \times 2^X \times S$ 是一个基于状态的迁移关系。

在服务交互过程中, $(s, m, \delta, \varphi, \gamma, s')$ 表示从状态 s 到状态 s' 的转换, m 是状态 s 接收或发送的消息, δ 是 X 上的一个时钟约束, φ 是 D 上的一个消息数据约束,它们在转换发生时被满足, $\gamma \subseteq X$ 是在该转换发生时复位零值的时钟集合。此外,在一个转换过程中涉及的消息数据值 $L(m, d)$ 是当前状态 s 接收或发送的消息 m 中包含的数据内容。

定义 5 两个数据和时间感知的服务模型 Σ_1 和 Σ_2 的组合模型 $\Sigma = \Sigma_1 \times \Sigma_2 = (S_1 \times S_2, s_{01} \times s_{02}, F_1 \times F_2, M_1 \cup M_2, X_1 \cup X_2, D_1 \cup D_2, I, I', R)$, 其中 $((s_1, s_2), m, \delta, \varphi, \gamma, (s_1', s_2')) \in R$ 当且仅当:

- $(s_1, m, \delta, \varphi, \gamma, s_1') \in R_1, m \notin M_2, s_2' = s_2$, 或
- $(s_2, m, \delta, \varphi, \gamma, s_2') \in R_2, m \notin M_1, s_1' = s_1$, 或
- $(s_1, m, \delta_1, \varphi_1, \gamma_1, s_1') \in R_1, (s_2, m, \delta_2, \varphi_2, \gamma_2, s_2') \in R_2,$

$m \in M_1, m \in M_2, \delta = \delta_1 \wedge \delta_2, \varphi = \varphi_1 \wedge \varphi_2, \gamma = \gamma_1 \cup \gamma_2$ 。

可以看出,通过对有限自动机的扩展,数据和时间感知的服务模型 DTSM 具有了表达数据流信息和时间信息的能力。另外,为了更好地关注于数据流信息和时间信息的表达,我们忽略了一些细节问题,比如服务间的消息通信、数据流信息的存储和因数据流信息而造成的状态爆炸等问题。

4 Web 服务组合模型检测

4.1 服务组合规约至 DTSM 的转换

BPEL(Business Process Execution Language) 是当前使用最广泛的服务组合描述语言,它通过具有控制语言构成的“流程-集成-类型”机制,将一组已有的服务定义为一个新的 Web 服务。本节介绍服务组合描述语言 BPEL 至数据和时间感知的服务模型 DTSM 的转换规则。

图 2 展示了部分典型的 BPEL 语句至 DTSM 的转换。其中 $\langle \text{receive} \rangle$ 活动可以用来接收来自外部的消息,这个活动可以通过一个迁移动作来完成; $\langle \text{invoke} \rangle$ 活动则用于调用一

个 Web 服务,它需要通过两个迁移动作来完成,第一个迁移动作用于发送一个消息。第二个迁移动作用于接收一个消息,如果接收到错误消息,则需要进行错误处理。以上都属于基本活动,它们是和服务进行交互的最简单的形式,而 $\langle \text{repeatUntil} \rangle$ 、 $\langle \text{while} \rangle$ 和 $\langle \text{sequence} \rangle$ 活动则属于结构化活动。结构化活动规定了活动的运行方式,通过将基本活动结构化来创建业务流程。它们内部定义了多个其它活动,因此在 DTSM 中需要进行多个迁移动作才能完成。

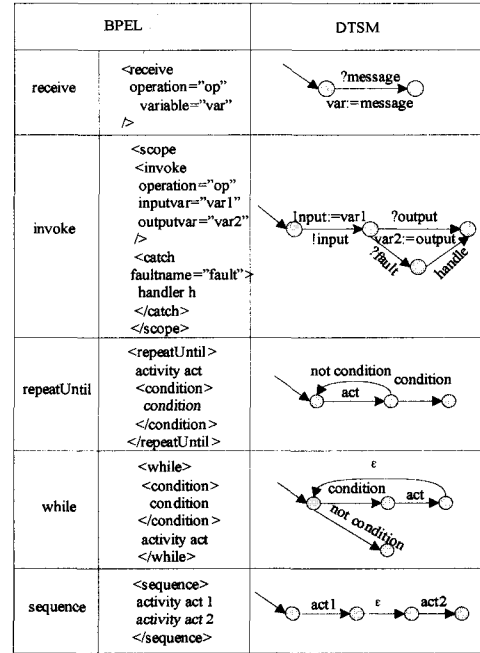


图 2 典型的 BPEL 至 DTSM 的转换

图 3 展示了部分具有时间信息和数据流信息的 BPEL 语句至 DTSM 的转换。其中 $\langle \text{pick} \rangle$ 活动和 $\langle \text{wait} \rangle$ 活动都含有对时间信息的表达,在 DTSM 中用含有时钟变量 x 的表达式来表示对时间的约束。在 $\langle \text{pick} \rangle$ 活动中选择了要执行的事件后,需要等到 time 时间之后才能执行活动,在 DTSM 中表现为时钟变量 $x > \text{time}$ 满足后才能完成一次状态迁移动作。

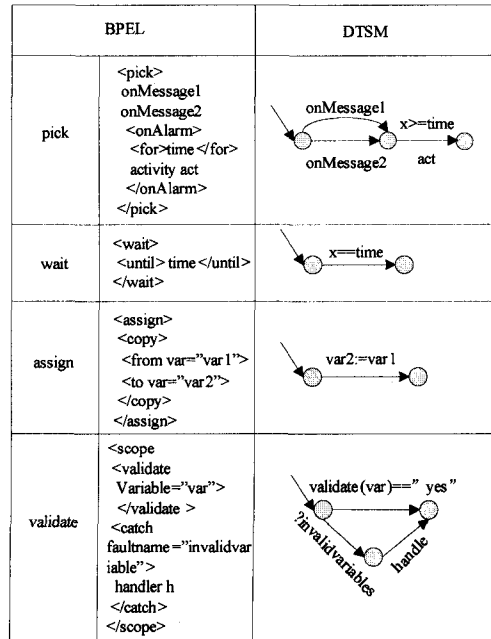


图 3 具有时间和数据流信息的 BPEL 至 DTSM 的转换

另外,〈assign〉活动和〈validate〉活动含有对数据流信息的表达,〈assign〉活动具有复制值的能力,可以用来在变量之间拷贝消息数据,实现对数据的操作。〈validate〉活动可以有效验证变量,当验证满足时,才能继续往下执行,否则将会抛出 invalidVariables 错误,在 DTSM 中将根据验证的不同情况执行对应的迁移动作。

4.2 DTSM 至 UPPAAL 规约的转换

基于时间自动机的模型检测工具 UPPAAL 是一个用于实时系统建模、模拟与验证的工具,由瑞典 Uppsala 大学与丹麦 Aalborg 大学联合研发,其主要优点是高效性和实用性。与其它模型检测工具(如 Spin、NuSMV 等)相比,UPPAAL 的输入语言与 DTSM 最为相近,故本文选用 UPPAAL 作为验证 Web 服务组合属性的工具。

由于 DTSM 与 UPPAAL 规约的语义具有相似性,部分元素(如状态上的不变式、迁移上的卫士等)之间可以直接做映射,但还需要考虑对时间信息和数据流信息表达的转换。对于时间信息的表达,由于 UPPAAL 本身基于时间自动机,具有表达时间信息的能力,而且 DTSM 中对时间的解释也是基于时间自动机中的时钟概念,故两者对时间的表达类似,无需进行转换。

下面考虑对数据流信息表达的转换。在 DTSM 中,是采用消息数据的概念来表达数据流信息,而 UPPAAL 规约并不直接支持对消息数据的表达,为此我们定义一个转换函数: $Trans: L(m, d) \rightarrow Var$, 其中 $m \in M$ 为消息的集合, $d \in D$ 为消息数据变量的集合, Var 为整型变量的集合。这样我们就可以将消息中包含的消息数据直接用整型变量标识。

由于消息数据包含多种类型的数据变量,如整型型数据变量、实数型数据变量和字符串型数据变量,而 UPPAAL 规约并不支持实数型数据变量和字符串型数据变量。为此,可将字符串型消息数据值用大型的特殊整数(如 32768)代替即可。而对于实数型消息数据,在精度要求不高的情况下可用近似整型值代替。

4.3 实例分析

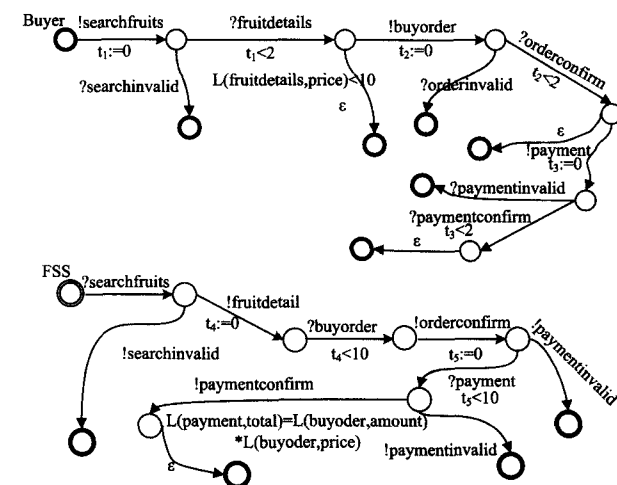


图4 购买者 Buyer 和 FSS 的 DTSM 模型

以水果销售服务 FSS 为例,假设在交互过程中,购买者 Buyer 向水果销售服务 FSS 发出消息后必须在 2 个单位时间内收到 FSS 的反馈消息,而 FSS 经过 10 个单位时间仍未收到购买者 Buyer 响应的情况下就会放弃交易。另外,购买者 Buyer 对所查询到的水果价格在 10 元以内才会向 FSS 发送

订购单,而 FSS 对于收到的支付订单,需要验证总的支付金额 total 正确后才会发送订购单确认消息。

根据以上需求,购买者 Buyer 和水果销售服务 FSS 的 DTSM 模型如图 4 所示。

根据上述转换规则,将购买者 Buyer 和服务 FSS 的 DTSM 模型转换至 UPPAAL 规约,其结果如图 5 所示。

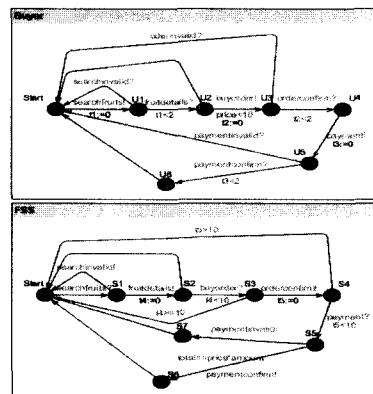


图5 购买者 Buyer 和 FSS 的 UPPAAL 规约

下面使用模型检测工具 UPPAAL 对该组合服务的部分行为属性、数据属性和时间属性进行分析与验证,验证结果如图 6 所示。

1)行为属性:在任何时刻,不可能出现在购买者 Buyer 已经支付了账单的情况下,服务 FSS 却取消了购买者的订购单。该性质可表示为:

$$A[] \text{not}(\text{Buyer. U3 and FSS. S7})$$

2)数据属性:只有当购买者 Buyer 支付了正确的金额数目时,服务 FSS 才会发送支付确认信息。该性质可表示为:

$$A[] \text{FSS. S6 imply}(\text{amount} * \text{price} == \text{total})$$

3)时间属性:购买者 Buyer 提交订购单后,服务 FSS 必须在 2 个单位时间内给出订购单确认信息。该性质可表示为:

$$E[] \text{Buyer. U6 imply}(t_3 < 2)$$

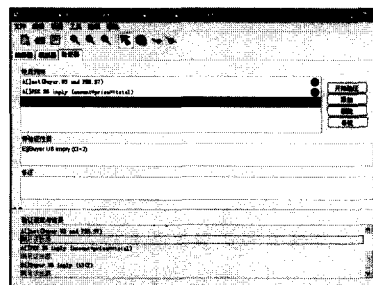


图6 属性验证结果

结束语 本文对有限自动机进行扩展,提出了一种基于数据和时间感知的服务模型 DTSM 的 Web 服务组合模型检测方法。结合 FSS 实例,研究了对服务组合的行为属性、数据属性和时间属性的分析与验证。我们今后将对 DTSM 作进一步扩展,以增强该模型的表达能力,同时考虑采用限界模型检测技术缓解因消息数据过多而产生的状态爆炸等问题。

参考文献

- [1] Clarke E M, Grumberg O, Peled D A. Model Checking[M]. Cambridge, UK: MIT Press, 1999

(下转第 136 页)

$$\begin{aligned} \text{SYS}_1^1 &= /exception^{\circ}(exception | /L^{\circ}(alt.(send_{warning}.alt. \\ &\quad (send_{cwc}.d_0 | 1) | 1) | Resource_{\alpha_0} | Handler_{cwc})) \\ \text{SYS}_2^2 &= /exception^{\circ}(exception | /L^{\circ}(warning | alt.(send_{cwc}.d_0 \\ &\quad | 1) | /token^{\circ}(d_{\alpha_01} | alt.(send_{token}.d_{\alpha_00} | 1) | Wor- \\ &\quad kitem_{\alpha_0}) | Handler_{cwc})) \\ \text{SYS}_3^3 &= /exception^{\circ}(exception | L^{\circ}(warning | cwc | d_0 | / \\ &\quad token^{\circ}(d_{\alpha_01} | alt.(send_{token}.d_{\alpha_00} | 1) | Workitem_{\alpha_0}) \\ &\quad | d_1)) \\ \text{SYS}_4^4 &= /exception^{\circ}(exception | /L^{\circ}(warning | cwc | d_0 | / \\ &\quad token^{\circ}(token | d_{\alpha_01} | d_{\alpha_00} | d_{\alpha_02}) | d_1)) \end{aligned}$$

该序列表示系统抛出违例(通过 *exception* 链接),系统变为 SYS_1^1 ;之后通过 *warning* 链接通知资源进行工作项级违例处理,系统变为 SYS_2^2 (由于 OCO 模式不对工作项产生实际影响,故剩余工作为: d_{α_01} ,详细动作可根据实际情况细化);系统再通过 *cwc* 链接触发案例级别违例处理,系统变为 SYS_3^3 (具体工作为: d_0 ,详细动作可根据实际情况细化);最后,系统继续执行工作项(通过 *token* 触发工作项,具体工作为: d_{α_02})。根据 OCO 模式的描述,本文中的 $Resource_{\alpha_0}$ 将违例的通知与工作项的触发设计为并行方式,所以系统中资源获得违例通知与工作项的触发没有先后顺序。由此其他 3 条变迁序列分别为:

- 系统 $\text{SYS}_{\alpha_0-cwc-nil}$ 先触发工作项(系统变迁至 SYS_1^1 状态),之后再进行违例的抛出、捕获,最后执行后续违例处理(具体包含工作项级违例处理和案例级违例处理);

- 系统也可能在捕获违例之后,直接触发工作项(系统变迁至 SYS_2^2 状态),之后再进行后续违例处理(具体包含工作项级违例处理和案例级违例处理);

- 系统捕获违例,完成工作项级别违例处理,在执行案例级违例处理以前,进行工作项的触发(系统变迁至 SYS_3^3 状态),之后再进行案例级违例处理。

最后,由于 OCO-CWC-NIL 模式不要求进行流程补偿处理,因此上述模型的动态分析未涉及违例处理中的补偿处理。

结束语 本文首先介绍了流程的违例及其分类;针对已知违例的情况,论文使用 CCS 偶图对流程工作项级、案例级违例处理,以及相关补偿处理进行分析和表述;基于违例处理策略的讨论,本文总结了违例处理模式的基本形式,并通过 CCS 偶图的动态特征分析了违例处理的动态行为。

未来的研究工作可以从以下几个方面展开:首先,针对流程中的未知违例进行研究,总结未知违例的技术特点;其次,对未知违例的处理提出新的处理策略和方法;最后,将违例处理的策略和方法应用于灵活的流程管理应用系统中。

参考文献

- [1] Russell N, van der Aalst W M P, ter Hofstede A H M. Workflow Exception Patterns[C]//Proceedings of the 18th International Conference on Advanced Information Systems Engineering(CAISE 06). vol. 4001 of LNCS, Berlin: Springer-Verlag, 2006:288-302
- [2] Milner R. The space and motion of communicating agents[M]. Cambridge University Press, April 2009
- [3] Milner R. Calculi for interaction[J]. Acta Informatica, 1996: 707-737
- [4] Eder J, Liebhart W. The workflow activity model (WAMO)[C]//Proceedings of the 3rd international conference on Cooperative Information Systems (CoopIS). Vienna, Austria, May 1995
- [5] Rinderle S, Reichert M, Dadam P. Correctness criteria for dynamic changes in workflow systems-a survey [J]. Data and Knowledge Engineering, 2004, 50: 9-34
- [6] Grohmann D, Miculan M. Controlling resource access in Directed Bigraphs[C]//Proceedings of the 7th International Workshop on Graph Transformation and Visual Modeling Techniques (GT-VMT 2008), In Electronic Communications of the EASST (Vol. 10), European Association of Software Science and Technology
- [7] Conforti G, Macedonio D, Sassone V. Bigraphical Logics for XML[C]//Proceedings of the 13th Italian Symposium on Advanced Database Systems (SEBD). 2005:392-399
- [8] Bundgaard M, et al. Formalizing WS-BPEL in Binding Bigraphs [R]. May 2008. <http://espen.dk/papers/BPELinBindingBigraphsWSFM.pdf>
- [9] Birkedal L, Bundgaard M, Damgaard T C, et al. Bigraphical programming languages for pervasive computing[C]//Proceedings of the International Workshop on Combining Theory and Systems Building in Pervasive Computing. 2006:653-658
- [10] Jensen O H, Milner R. Bigraphs and mobile processes (revised) [R]. UCAM-CL-TR-580. University of Cambridge Computer Laboratory, 2004
- [11] Hildebrandt T, Niss H, Olsen M. Formalising business process execution with bigraphs and reactive XML[C]//Proceedings of the 8th International Conference on Coordination Models and Languages. vol. 4038 of LNCS, Springer Verlag, 2006:113-129
- [12] Milner R. A Calculus of Communicating Systems [J]. LNCS, 1980, 94
- [13] 巫茜, 张栋, 包坤. 基于角色的工作流平台访问控制安全模型 [J]. 重庆理工大学学报:自然科学版, 2011, 25(3): 78-82
- (上接第 126 页)
- [2] Fu X, Bultan T, Su J. Analysis of Interacting BPEL Web Service [C]//Proc. of the 13th International Conference on the World Wide. New York, USA, 2004:621-630
- [3] Diaz G, Pardo J J, Cambronero M E, et al. Automatic Translation of WS-CDL Choreographies to Timed Automata [C] //Proc. of the International Workshop on Web Services and Formal Methods. LNCS, 3670. Versailles, France, 2005:230-242
- [4] Zhao Xiang-peng, Yang Hong-li, Qiu Zong-yan. Towards The Formal Model and Verification of Web Service Choreography Description Language [C]//Proc. of the 3th International Workshop Web Services and Formal Method (WS-FM 2006). 2006: 273-287
- [5] Guermouche N, Godart C. Timed Model Checking Based Approach for Web Services Analysis [C]//Proc. of the 7th Int'l Conf. on Web Services. Los Angeles, USA, 2009:213-221
- [6] 门鹏, 段振华. 着色 Petri 网模型检测工具的扩展及其 Web 服务组合中的应用 [J]. 计算机研究与发展, 2009, 46(8): 1294-1303
- [7] 骆翔, 字陈艳, 古天龙, 等. 基于时态认知逻辑的 Web 服务模型检测 [J]. 计算机科学, 2009, 36(8): 153-157
- [8] 何亚丽, 戎玫, 张广泉. 一种基于 UPPAAL 的 Web 服务组合模型检测方法 [J]. 计算机科学, 2010, 37(11): 122-125