

是否满足行为等价.之所以选择弱互模拟关系作为行为等价检测的标准,而不采用迹等价或强互模拟关系,原因有二:一是,映射产生的 workflow 网中具有 τ 变迁,对观察者而言,该变迁是透明的、不可见的.例如,将并行分叉网关和并行汇聚网关映射产生的 Petri 网片段中便会出现 τ 变迁.由于强互模拟关系无法比较不可见动作,因此,本文不采用强互模拟关系.二是,迹等价不区分选择的时机,会将不确定性的可达图等价于确定性的可达图,未考虑编排在某状态下的所有分支结构.因此,本文也不采用迹等价.

针对表 1 中列举的编排及其映射生成的 workflow 网,我们通过人工分析的方式对映射规则进行了一致性测试.测试结果表明,第 3 节提出的映射规则是一致的.

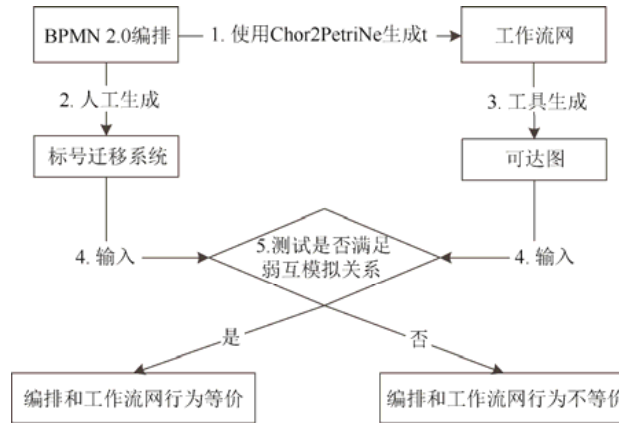


Fig. 13 Testing conformance of the mapping rules

图 13 测试映射规则的一致性

6 相关工作

相关工作将从 BPMN 标准规约中编制与编排的形式语义定义、编排建模语言的形式语义定义及编排的语义分析这 3 个方面进行对比.

6.1 编制与编排的形式语义定义

BPMN 2.0 标准规约使用自然语言定义了编制的执行语义^[1].但是,用自然语义定义的编制的执行语义,无法直接实现为工具,用于对编制进行模拟、验证和执行^[12,13].为此,文献[2,12–18]使用不同的形式化方法定义了 BPMN 编制的形式语义.

文献[2]使用 Petri 网形式定义了 BPMN 1.0 编制子集的语义.它提出了 BPMN 中建模符号到 Petri 网元素的映射关系.这样做的好处是:可以使用 Petri 网丰富的分析技术对 BPMN 进行合理性检测.

文献[12]使用图重写规则(graph rewriting rule)形式定义了 BPMN 2.0 编制子集的执行语义.该文献建立了 BPMN 2.0 标准中非形式定义编制的执行语义与用图重写规则形式定义的编制的执行语义间的直接联系.这样做的好处是:每个形式定义规则的正确性都可以通过 BPMN 2.0 标准中非形式的规约来验证.此外,该形式化的执行语义还可视为一个测试套件,用来测试实现 BPMN 执行语义的工具的一致性.作为该文献工作的扩展,文献[13]使用图重写规则形式定义了更大的 BPMN 2.0 编制子集的执行语义.

文献[14,15]使用进程代数 CSP 形式定义了 BPMN 1.0 编制子集的语义.这样做的好处是:建模者可以使用形式语义检测不同抽象层次业务流程模型间的一致性.具体而言,一方面,建模者可以对业务流程模型领域相关的时序性质进行检测,例如订单提交后,一个响应必须在 24 小时内发送到客户端;另一方面,建模者可以对业务流程模型的通用性质进行检测,例如:无死锁(deadlock-freeness)和正常结束(proper completion).通常,在业务流程管理领域,把对无死锁性质和正常结束性质的检测称为合理性检测(soundness checking).

文献[16]使用 YAWL 定义了 BPMN 1.0 编制子集的语义.它提出了 BPMN 元素到 YAWL 元素的映射关系.

这样做的好处是:建模者可以使用 YAWL 的验证技术对业务流程模型的合理性和活性进行检测。

文献[17]使用进程代数 COWS(calculus of orchestration of Web services)形式定义了 BPMN 1.0 编制子集的语义.这样做的好处是:基于形式语义,建模者不仅可以对业务流程模型进行合理性检测,还可以对业务流程模型(只要业务流程模型具有模拟信息)进行定量模拟.与文献[2,14,15]相比,该文献从控制流和数据流两个方面定义了 BPMN 1.0 子集的语义.

文献[18]使用线性时序逻辑 LTL(linear temporal logic)形式定义了 BPMN 1.2 编制子集的执行语义.这样做的好处是:一方面,在 LTL 框架下,建模者可以对业务流程模型的时序性质进行检测;另一方面,还可以为 BPMN 1.2 标准提供形式的语义规约.

文献[19]提出了一种模型转换的方法,将企业流程建模语言 TiPLM(Tsinghua InfoTech product lifecycle management solution)定义的工作流程转换为行为等价的 Petri 网,并使用 WoPeD 对 Petri 网进行了可靠性验证.TiPLM 是一种类似 BPMN 编制的流程建模语言.

与上述文献相比,本文关注的是 BPMN 2.0 编排的形式语义定义,而非编制的形式语义定义.需要特别注意的是,BPMN 1.X(BPMN 1.0、BPMN 1.1 和 BPMN 1.2)中只有编制模型的概念,没有编排模型.表 3 从对象、方法、用途、范围这 4 个方面,将本文工作与上述工作进行了对比和总结.本文定义的形式语义不仅可作为 BPMN 2.0 编排的语义规约,用于消除二义性和不一致,还可作为形式化分析的基础,支持编排的语义分析.具体比较如下.

与文献[12-17]相比,本文所做工作的区别在于:(1) 关注对象.本文关注的建模语言是 BPMN 2.0 编排,而这些文献关注的建模语言是编制.(2) 使用的形式化方法不同.本文使用 Petri 网定义编排的形式语义,而这些文献使用的是其他形式化方法定义编制的形式语义,如 CSP、YAWL、COWS、LTL、graph rewriting rule.

Table 3 Semantics of BPMN orchestrations

表 3 BPMN 编制的语义

文献	对象	方法	用途	范围
BPMN 2.0 标准 ^[1]	BPMN 2.0 的编制	自然语言	语义规约	全部
文献[2]	BPMN 1.0 的编制	Petri nets	语义规约,合理性检测	控制流子集
文献[12,13]	BPMN 2.0 的编制	Graph rewriting rule	语义规约,一致性检测	控制流子集
文献[14,15]	BPMN 1.0 的编制	CSP	时序性质检测,合理性检测	控制流子集
文献[16]	BPMN 1.0 的编制	YAWL	合理性检测,活性检测	控制流子集
文献[17]	BPMN 1.0 的编制	COWS	合理性检测,定量模拟	控制流子集,数据流子集
文献[18]	BPMN 1.2 的编制	LTL	时序性质检测,语义规约	控制流子集
文献[19]	TiPLM 的工作流程	Petri nets	合理性检测	控制流子集
本文	BPMN 2.0 的编排	Petri nets	语义规约,语义分析	控制流子集

与本文工作最为相似的是文献[2]和文献[19]所做的工作.文献[2]使用 Petri 网形式定义了编制的语义,与之相比,本文所做工作的区别在于:(1) 本文定义开始事件、结束事件、中间事件、排他数据(事件)网关和并发网关到 Petri 网片段的映射方式与文献[2]不同;(2) 文献[2]使用库所融合的方式将映射得到的 Petri 网模块组合为一个 Petri 网,而本文使用流关系将映射得到的 Petri 网片段组合为一个 Petri 网.这样做的好处是,本文映射产生的 Petri 网具有更少的库所.(3) 文献[2]中没有单独定义流关系到 Petri 网片段的映射,而本文考虑了这一点.文献[19]使用 Petri 网形式定义了 TiPLM 工作流程的形式语义,并使用 WoPeD 对 Petri 网进行了合理性验证,与之相比,本文所做工作的区别在于:(1) 在 TiPLM 工作流程中,建模元素有 4 种:开始事件、结束事件、任务和网关;而本文定义的编排中,建模元素有 5 种:开始事件、结束事件、编排活动、中间事件和网关,其中,编排活动包含编排任务和调用编排;(2) TiPLM 工作流程没有层次任务(hierarchical task),不考虑层次性;而本文定义的编排可以具有层次结构的子编排.

6.2 编排建模语言的形式语义定义

为定义编排,大量编排建模语言被提了出来,例如:BPMN 2.0^[1]、协作图(collaboration diagram,简称 CD)^[20]、Web 服务编排规约语言(Web services choreography description language,简称 WS-CDL)^[3]、Let's Dance^[21]、IPN(interaction Petri nets)^[22]、会话协议(conservation protocol,简称 CP)^[23].这些编排建模语言可分为两类:具有

形式语义的编排建模语言和不具有形式语义的编排建模语言.具有形式语义的编排建模语言包括:CD、IPN 和会话协议.针对不具有形式语义的编排建模语言,文献[1,24–26]使用不同的方法定义了编排的形式语义.

为了支持编排的定义,与 BPMN 1.X(BPMN 1.0、BPMN 1.1 和 BPMN 1.2)相比,BPMN 2.0 首次提出了编排的概念,将其视为“头等公民”^[1].BPMN 2.0 标准规约中用自然语言描述了 BPMN 2.0 编排的语义,但是用自然语言描述的语义存在二义性和不一致.

文献[24]使用 Pi 演算定义了 Let's Dance 的形式语义.这样做的好处是:基于形式语义,建模者可以把对编排的可达性(reachability)进行分析的问题规约为判断两个 Pi 演算进程间是否满足弱互模拟关系的问题.

文献[25]使用时间自动机(timed automata)形式定义了 WS-CDL 子集的操作语义.这样做的好处是:基于操作语义,建模者可以使用 UPPAAL 工具来对编排进行各种性质的检测,例如:不变量、可达性、前条件和后条件(pre and post conditions)、隐含属性(implication properties)和时间限制(time restriction).

文献[26]针对互连式编排建模存在的不足,借鉴交互式编排建模的思想,将 BPMN 1.1 扩展为 iBPMN,并使用交互式 Petri 网(IPN)形式定义了 iBPMN 的执行语义.这样做的好处是:一方面,建模者可以避免对编排解释的二义性;另一方面,可以使用 Petri 网的分析技术对编排模型进行合理性分析.

文献[27]针对 BPMN 2.0 编排,使用进程代数 LOTOS NT 定义了其形式语义.这样做的好处是:基于形式语义,可以对 BPMN 2.0 进行可实现性(realizability)分析.

与上述文献相比,本文关注的是 BPMN 2.0 编排的形式语义定义.表 4 从编排建模语言、方法、用途、范围这 4 个方面,将本文工作与上述工作进行了对比和总结.文献[24–26]分别针对的编排建模语言是:Let's Dance、WS-CDL、iBPMN,而本文关注的是 BPMN 2.0 编排.具体比较如下.

与文献[24–26]相比,本文所做工作的区别在于:关注的编排建模语言不同.本文关注的编排建模语言是 BPMN 2.0 编排,而这些文献分别关注的是其他编排建模语言:Let's Dance、WS-CDL 和 iBPMN.

与本文工作最为相似的是文献[1]和文献[27]所做的工作,文献[1]使用自然语言描述 BPMN 2.0 编排的语义,会产生二义性和不一致;而本文使用 Petri 网,准确定义了 BPMN 2.0 编排的语义.文献[27]使用进程代数 LOTOS NT 定义了 BPMN 2.0 编排的语义,而本文使用 Petri 网定义 BPMN 2.0 编排的语义.与文献[27]所做工作的区别在于:(1) 文献[27]定义编排语义时,限制网关必须是平衡的,即若一个网关有 n 个输入分支,则应有一个对应的网关有 n 个输出分支,本文无此限制;(2) 文献[27]没有考虑构造结构的多样性.(3) 文献[27]未考虑标准循环的编排任务和子编排的形式语义定义.

Table 4 Semantics of choreographies

表 4 编排的语义

编排的语义	编排建模语言	方法	用途	范围
BPMN 2.0 标准 ^[1]	BPMN 2.0	自然语言	语义规约	全部
文献[24]	Let's Dance	Pi calculus	可达性检测	控制流子集
文献[25]	WS-CDL	Timed automata	不变量 可达性 前条件和后条件隐含属性 时间限制	控制流子集
文献[26]	iBPMN	Petri nets	语义规约 合理性检测	控制流子集
文献[27]	BPMN 2.0 编排	进程代数 LOTOS NT	语义分析	控制流子集
本文	BPMN 2.0 编排	Petri nets	语义规约 语义分析	控制流子集

6.3 编排的语义分析

编排作为参与者协同的全局规约,需将其合成为每个参与者,才能用于指导参与者的实现.但是,合成的参与者未必总能准确实现编排^[28].进一步地,给定一个编排,能否将全局编排映射为每个参与者并确保这些参与者间的交互精确匹配流程编排,这就是可实现性分析的内涵^[4].为此,文献[4,22,23,29–34]对编排进行了可实现性分析.这些方法大致可以分为 3 类:基于自动机的可实现性分析方法、基于进程代数的可实现性分析方法和基

于 Petri 网的可实现性分析方法.

基于自动机的可实现性分析方法.针对基于会话协议定义的编排,文献[23]以自动机作为形式化基础,最早提出了会话协议可实现需满足的3个充分条件:无损连接(lossless join)、同步兼容(synchronous compatible)、自治性(autonomous).在此基础上,文献[29-31]针对会话协议的子类,提出了同步性条件(synchronizability conditions),并将会话协议的可实现性分析转换为对同步性条件的检验.文献[32]针对任意发起者的会话协议(arbitrary-initiator protocols),提出了一个新的充分条件,用于检验会话协议的可实现性.文献[33]提出了会话协议可实现的充分必要条件,将可实现性分析转换为对同步性和合适性(well-formedness)的检测,从而可使用现有的行为等价检测工具和模型检测工具,对会话协议的可实现性进行自动检测.

基于进程代数的可实现性分析方法.针对基于协作图定义的编排子类,文献[4]以进程代数 LOTOS NT 作为形式化基础,提出一种将协作图定义的编排形式转换为基于 LOTOS NT 定义的进程表达式的方法,从强互模拟的角度,检验了同步通信和异步通信环境下协作图的可实现性.针对 BPMN 2.0 中的编排子类,文献[27]以进程代数 LOTOS NT 作为形式化基础,提出一种把编排形式转换为 LOTOS NT 定义的进程表达式的方法,从强互模拟的角度,对同步通信环境和异步通信环境下编排的可实现性进行检验.在此基础上,文献[34]提出一种模块化的框架,用于自动验证 WSDL、会话协议和 BPMN 2.0 编排的可实现性.

基于 Petri 网的可实现性分析方法.文献[22]以 Petri 网作为形式化基础,用 IPN 定义编排,并从分支互模拟的角度,对同步通信环境下 IPN 的可实现性进行了分析.但 Internet 环境下存在网络延迟,因此必须对异步通信环境下编排的可实现性进行分析.但是,该方法只能分析同步通信环境下 IPN 的可实现性,不能对异步通信环境下 IPN 的可实现性进行分析.

与上述文献相比,本文的语义分析更关注从控制流的角度,分析 BPMN 2.0 编排表现出由面向图流程定义产生的语义错误,如流增加、流减少、死编排任务等.这样做的原因在于:从控制流方面,确保编排的语义正确性是讨论可实现性的前提.因为,可实现性分析本质上是讨论整体编排与局部参与者间的语义一致性.若整体编排的正确性无法保证,则讨论整体编排与局部参与者间的语义将无任何实际意义.此外,本文的语义分析还考虑了 BPMN 2.0 标准规约中对编排的结构约束:编排活动序列的基本规则.

7 结束语

BPMN 2.0 标准规约中编排缺少形式语义及相应的分析技术,这将阻碍对 BPMN 2.0 编排的语义分析.首先,通过建立 BPMN 2.0 编排到工作流网的映射,使用 Petri 网准确定义编排的语义;其次,借助 Petri 网的分析技术,把 BPMN 2.0 编排中存在的语义错误规约为工作流网的结构问题或性质问题,对 BPMN 2.0 编排进行语义分析;最后,通过实验表明,这种形式化可以识别 BPMN 2.0 编排中存在的错误.

本文并没有考虑复杂网关和包含网关的语义定义.下一步的工作重点是:提出 BPMN 2.0 编排到参与者的映射算法、对 BPMN 2.0 编排进行可实现性分析以及根据 Petri 网中发现的问题,自动地将其对应到原编排.

References:

- [1] OMG. Business Process Model and Notation (BPMN) Version 2.0. 2011. <http://www.omg.org/spec/BPMN/2.0/>
- [2] Dijkman RM, Dumas M, Ouyang C. Semantics and analysis of business process models in BPMN. *Information & Software Technology*, 2008,50(12):1281-1294. [doi: 10.1016/j.infsof.2008.02.006]
- [3] W3C. Web Service Choreography Description Language (WS-CDL). 2005. <http://www.w3.org/TR/ws-cdl-10/>
- [4] Salaün G, Bultan T, Roohi N. Realizability of choreographies using process algebra encodings. *IEEE Trans. on Services Computing*, 2012,5(3):290-302. [doi: 10.1109/TSC.2011.9]
- [5] Petri CA. Kommunikation mit automaten [Ph.D. Thesis]. Institut für Instrumentelle Mathematik, Schriften des IIM 2, 1962.
- [6] Yuan CY. The Principle and Application of Petri Nets. Beijing: Electronic Industry Publishing House, 2005 (in Chinese).
- [7] van der Aalst WMP. The application of Petri nets to workflow management. *Journal of Circuits, Systems, and Computers*, 1998,8(1):21-66. [doi: 10.1142/S0218126698000043]

- [8] Billington J, Christensen S, van Hee KV, Kindler E, Kummer O, Petrucci L, Post R, Stehno C, Weber M. The Petri net markup language: Concepts, technology, and tools. In: Proc. of the Int'l Conf. on Applications and Theory of Petri Nets. Berlin: Springer-Verlag, 2003. 483–505. [doi: 10.1007/3-540-44919-1_31]
- [9] Eckleder A, Freytag T. WoPeD 2.0 goes BPEL 2.0. In: Proc. of the German Workshop on Algorithms and TOOLS for Petri Nets, Algorithmen Und Werkzeuge Für Petrinetze, Awpn 2008. Rostock, 2008. 75–80.
- [10] BPM academic initiative. <http://bpt.hpi.uni-potsdam.de/BPMAcademicInitiative/>
- [11] Milner R. Communicating and Mobile Systems: The Pi-Calculus. Cambridge: Cambridge University Press, 1999.
- [12] Dijkman R, Gorp PV. BPMN 2.0 execution semantics formalized as graph rewrite rules. Lecture Notes in Business Information Processing, 2010,67:16–30. [doi: 10.1007/978-3-642-16298-5_4]
- [13] Gorp PV, Dijkman R. A visual token-based formalization of BPMN 2.0 based on in-place transformations. Information & Software Technology, 2013,55(5):365–394. [doi: 10.1016/j.infsof.2012.08.014]
- [14] Wong PYH, Gibbons J. A process semantics for BPMN. In: Proc. of the Int'l Conf. on Formal Engineering Methods. Berlin: Springer-Verlag, 2008. 355–374. [doi: 10.1007/978-3-540-88194-0_22]
- [15] Wong PYH, Gibbons J. Formalisations and applications of BPMN. Science of Computer Programming, 2011,76(8):633–650. [doi: 10.1016/j.scico.2009.09.010]
- [16] Ye JH, Sun SX, Song W, Wen L. Formal semantics of BPMN process models using YAWL. In: Proc. of the Int'l Symp. on Intelligent Information Technology Application. Washington: IEEE, 2008. 70–74. [doi: 10.1109/IITA.2008.68]
- [17] Prandi D, Quaglia P, Zannone N. Formal analysis of BPMN via a translation into COWS. In: Proc. of the 10th Int'l Conf. on Coordination Models and Languages. Berlin: Springer-Verlag, 2008. 249–263. [doi: 10.1007/978-3-540-68265-3_16]
- [18] Lam VSW. A precise execution semantics for BPMN. Int'l Journal of Computer Science, 2012,39(1):20–33.
- [19] Zha H, Aalst WMPVD, Wang J, Wen L, Sun J. Verifying workflow processes: A transformation-based approach. Software & Systems Modeling, 2011,10(2):253–264. [doi: 10.1007/s10270-010-0149-9]
- [20] Bultan T, Fu X. Specification of realizable service conversations using collaboration diagrams. Service Oriented Computing and Applications, 2008,2(1):27–39. [doi: 10.1007/s11761-008-0022-7]
- [21] Zaha JM, Barros A, Dumas M, Hofstede AT. Let's Dance: A language for service behavior modeling. In: Proc. of the 14th Int'l Conf. on Cooperative Information Systems. Berlin: Springer-Verlag, 2006. 145–162. [doi: 10.1007/11914853_10]
- [22] Decker G, Weske M. Local enforceability in interaction Petri nets. In: Business Process Management. Berlin, Heidelberg: Springer-Verlag, 2007. 305–319. [doi: 10.1007/978-3-540-75183-0_22]
- [23] Fu X, Bultan T, Su J. Conversation protocols: A formalism for specification and verification of reactive electronic services. Theoretical Computer Science, 2004,328(1):19–37. [doi: 10.1016/j.tcs.2004.07.004]
- [24] Decker G, Zaha JM, Dumas M. Execution semantics for service choreographies. Lecture Notes in Computer Science, 2006,34(3): 163–177. [doi: 10.1007/11841197_11]
- [25] Cambroner ME, Díaz G, Valero V, Martínez E. Validation and verification of Web services choreographies by using timed automata. Journal of Logic & Algebraic Programming, 2011,80(1):25–49. [doi: 10.1016/j.jlap.2010.02.001]
- [26] Decker G, Weske M. Interaction centric modeling of process choreographies. Information Systems, 2011,36:292–312. [doi: 10.1016/j.is.2010.06.005]
- [27] Poizat P, Salaün G. Checking the realizability of BPMN 2.0 choreographies. In: Proc. of the 27th Annual ACM Symp. on Applied Computing. Riva del Garda, 2011. 1927–1934. [doi: 10.1145/2245276.2232095]
- [28] Roohi N, Salaün G, France V. Realizability and dynamic reconfiguration of chor specifications. Informatica: An Int'l Journal of Computing and Informatics, 2011,35(1):39–49.
- [29] Basu S, Bultan T. Choreography conformance via synchronizability. In: Proc. of the 20th Int'l Conf. on World Wide Web. Hyderabad, 2011. 795–804. [doi: 10.1145/1963405.1963516]
- [30] Basu S, Bultan T, Ouederni M. Synchronizability for verification of asynchronously communicating systems. In: Proc. of the 13th Int'l Conf. on Verification, Model Checking, and Abstract Interpretation. Philadelphia, 2012. 56–71. [doi: 10.1007/978-3-642-27940-9_5]

- [31] Güdemann M, Salaün G, Ouederni M. Counterexample guided synthesis of monitors for realizability enforcement. Lecture Notes in Computer Science, 2012,7561:238–253. [doi: 10.1007/978-3-642-33386-6_20]
- [32] Hallé S, Bultan T. Realizability analysis for message-based interactions using shared-state projections. In: Proc. of the 18th ACM SIGSOFT Int'l Symp. on Foundations of Software Engineering. Seoul, 2010. 27–36. [doi: 10.1145/1882291.1882298]
- [33] Basu S, Bultan T, Ouederni M. Deciding choreography realizability. In: Proc. of the 39th Annual ACM SIGPLAN-SIGACT Symp. on Principles of Programming Languages. Philadelphia, 2012,47(1):191–202. [doi: 10.1145/2103656.2103680]
- [34] Güdemann M, Poizat P, Salaün G, Dumont A. A framework for verifying choreographies choreographies. In: Proc. of the 16th Int'l Conf. on Fundamental Approaches to Software Engineering, Rome, 2013. 226–230. [doi: 10.1007/978-3-642-37057-1_16]

附中文参考文献:

- [6] 袁崇义.Petri 网原理与应用.北京:电子工业出版社,2005.



代飞(1982—),男,四川乐山人,博士,副教授,CCF 专业会员,主要研究领域为软件工程,业务过程管理.



莫启(1986—),男,博士,讲师,主要研究领域为业务过程管理.



赵文卓(1992—),女,学士,主要研究领域为业务过程管理.



李彤(1963—),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为软件工程.



杨云(1981—),男,博士,教授,CCF 专业会员,主要研究领域为机器学习,数据挖掘,智能软件工程.



周华(1963—),男,博士,研究员,博士生导师,主要研究领域为软件工程,系统分析与集成.