

BPEL 流程建模中的交叠模式分析与转换*

张 曼¹, 段振华^{1,2+}, 王小兵¹

¹(西安电子科技大学 计算理论与技术研究所, 陕西 西安 710071)

²(西安电子科技大学 ISN 国家重点实验室, 陕西 西安 710071)

Analysis and Transformation of Overlapped Patterns in BPEL Process Modeling

ZHANG Man¹, DUAN Zhen-Hua^{1,2+}, WANG Xiao-Bing¹

¹(Institute of Computing Theory and Technology, Xidian University, Xi'an 710071, China)

²(State Key Laboratory of Integrated Service Networks, Xidian University, Xi'an 710071, China)

+ Corresponding author: E-mail: zhhdian@mail.xidian.edu.cn

Zhang M, Duan ZH, Wang XB. Analysis and transformation of overlapped patterns in BPEL process modeling. *Journal of Software*, 2011, 22(11): 2684-2697. <http://www.jos.org.cn/1000-9825/4075.htm>

Abstract: During the derivation of the business process execution language (BPEL) from graph-oriented process modeling languages, there exist structures called overlapped patterns, which mix sequential and parallel structures together. The existing duplication approach lacks systematical analysis and formal presentations. In view of this problem, a transformation of UML activity diagrams to BPEL based on WF-nets is proposed. By choosing free choice WF-nets as the foundation of activity diagrams and by utilizing the synthesis rules of live and bounded free choice systems, two types of overlapped patterns are defined. For one type of pattern, the duplication method is formally represented, and the transformation equivalence is proved using the concurrent regular expressions of Petri nets. For the other type, the applicable range of the duplication method is described. This method presents a solution in formal representations to the transformation of overlapped patterns in BPEL modeling, and distinguishes the derivation of the block-oriented modeling languages from graph-oriented ones.

Key words: BPEL (business process execution language); business process modeling; free choice WF-net; synthesis rule; overlapped pattern

摘 要: 由图形化流程建模语言生成可执行的业务流程语言(business process execution language, 简称 BPEL)时, 对于源模型中顺序与并发结构交织的情况(称为交叠模式), 传统的复制相关活动方法缺少系统分析及形式化描述. 针对这一现状, 提出基于工作流网的 UML 活动图生成 BPEL 方法, 以自由选择工作流网作为活动图的理论基础, 利用活的、有界的自由选择网系统的合成规则, 定义合理的自由选择工作流网中的两种交叠模式, 针对其中一种给出复制相关活动的形式化转换方法, 并借助 Petri 网的并发正则表达式证明转换等价性, 说明另一种交叠模式中复制相关活动方法的适用范围. 针对 BPEL 流程建模及图形化流程语言生成块状语言过程中的交叠模式转换问题, 给出形式

* 基金项目: 国家自然科学基金(60910004, 60873018, 91018010, 61003078, 61003079, 61133001); 国家重点基础研究发展计划(973)(2010CB328102); 国家教育部博士点基金(200807010012); 中央高校基本科研业务费专项资金(JY10000903004); 综合业务网国家重点实验室基金(201102001)

收稿时间: 2010-10-12; 修改时间: 2011-03-21; 定稿时间: 2011-07-01

化的描述与解决方法。

关键词: BPEL; 商业流程建模; 自由选择工作流网; 合成规则; 交叠模式

中图法分类号: TP311 文献标识码: A

可执行的业务流程语言(business process execution language,简称 BPEL)^[1]作为 Web 服务领域事实上的流程定义与执行标准,得到了广泛的关注。然而,BPEL 的抽象级别不适用于业务流程的分析与设计,因为它的定义基于块状结构;此外,它包含较多技术细节,难以被业务人员理解和使用。相反地,业务流程分析师和设计者更容易接受高级别的抽象模式。例如,以 UML 活动图作为流程设计模型,更符合业务人员的思维和使用习惯。模型驱动的体系结构通过模型转换的思想为商业流程建模提供了解决方案。活动图作为平台无关模型,由商业专家设计;BPEL 作为平台相关模型,由技术专家实现和部署。通过活动图到 BPEL 的转换,建立两者之间的桥梁。

如果以控制流为关注点,面向图形化的语言与面向块状语言间的差异性转换时面临的主要问题。在面向图形化的活动图中,动作被抽象为节点,用表示控制流的边连接;在 BPEL 中,为了适应流程引擎解释执行,它的定义以面向块状为主。这使得两者间的转换不易处理。关于面向图形化的流程语言到 BPEL 的转换,已有很多论文展开研究。多数方法将控制流中的顺序与并发结构分割开来,分别处理。对于顺序部分,通常借鉴传统的技术,例如,将非结构化的流程图转换为结构化的技术,或自动机生成正则式的理论;对于并发的部分,利用 BPEL 中受限制的图形化特征处理。具体地,并发部分被直接映射到一个 BPEL 的 Flow 活动,其中可能包含表示活动间限制关系的 control links。但是,对于顺序与并发结构无法分开处理的情况(通常称为交叠模式),当前方法大多存在局限性。最常见的方法是复制活动图中若干活动,得到一个顺序和并发能够分开处理的流程模型。这种方法缺乏系统的分析和形式化的描述及转换的等价性证明,也没有研究讨论其适用范围。另一种方法将顺序并发混合的部分直接映射到 BPEL 的 Flow 活动上,这种方法的缺点也是显而易见的:首先,生成代码的可读性差,无法体现 BPEL 的流程结构;其次,如果顺序结构中有循环出现,此方法会失效。要系统地处理流程模型中的交叠模式,必须提出新的解决方案。

借助工作流网(workflow net,简称 WF-net)^[2]这一形式化模型,描述活动图中交叠模式的形成过程与转换方法:一方面,以自由选择的工作流网为活动图的形式化表达;另一方面,定义更适宜映射到 BPEL 的可分离工作流网。利用活的、有界的自由选择网系统的合成规则分析自由选择工作流网的结构,定义两种交叠模式,针对其中一种给出复制相关活动的形式化转换方法,并借助 Petri 网的并发正则表达式证明转换的等价性,对另一种交叠模式说明复制相关活动方法的适用范围。第一,与源模型到目标模型的直接映射相比,采用工作流网作为中介使转换具有更可靠的理论基础;第二,对处理交叠模式的复制相关活动方法,给出系统和形式化的描述,描述其适宜范围,并给出转换等价性的形式化证明。针对 BPEL 流程建模及图形化流程语言生成块状语言过程中的交叠模式转换问题给出形式化的描述与解决方法。

1 相关工作

早期的活动图到 BPEL 转换研究中有很多方法专注于处理控制流中非结构化的循环。Hauser 等人给出两种由活动图生成 BPEL 代码的算法^[3]。Koehler 等人利用编译原理中的技术定义活动图到 BPEL 的转换规则^[4]。接着,Koehler 和 Hauser 讨论了如何基于 continuation semantics 将可化简的非结构化循环模型转换为 BPEL^[5]。Zhao 等人提出了能够处理不可化简循环的 REL 方法^[6],在后续工作中,他们将 REL 方法扩展到非结构化的并发区域的转换^[7]。以上这些方法考虑的流程模型要么不包含并发结构^[4-6],要么限制并发的定义使其不影响顺序部分的转换^[3,7],但没有考虑顺序与并发交织的问题。

其他图形化流程模型生成 BPEL 的研究中也有方法考虑到交叠模式。Aalst 等人在文献[8]中定义了工作流网到 BPEL 的映射,由示例中生成的代码片段可知采用复制相关活动的方法处理交叠模式。Ouyang 等人提出将非结构化的流程模型映射到 BPEL 的 Event Handler^[9],这种映射能够处理交叠模式,但生成的代码因无法体现流程结构而缺乏可读性。他们在后续工作中,以一个包含交叠模式的流程模型为例说明生成 BPEL 的完整方法,交

叠模式被映射为包含多个 links 的 Flow 活动^[10].Hauser 等人通过将工作流图分解为区域树,非形式化地讨论了交叠模式的形成过程^[11],并粗略地提到采用复制活动的方法由交叠模式生成 BPEL.他们在后续工作中补充了对交叠模式的分析^[12],但提出的新转换方法可能生成不等价的 BPEL 流程.

由于采用网分解的方法证明转换等价性,本文的研究与基于工作流网分解的分析方法相关.李建强等人在文献[13]中定义增加时间信息的工作流网,通过将无循环的自由选择工作流网分解为一组变迁组件网,验证工作流模型的时间有界性.他们在后续工作中将分解对象扩展到包含循环的自由选择工作流网,在此基础上提出了分析工作流模型性能的方法^[14].同样是对时间性能的分析,葛季栋等人提出一种基于 T -不变量的工作流网分解方法^[15],将分解所得的一组变迁组件网映射到 PERT 图以进行项目进度规划.庞善臣等人提出了基于 P -不变量的多项式分解算法^[16],降低了工作流网的时间性能分析方法的复杂度.以上研究为了分析时间性能,对工作流的分解都以整个网为对象.

2 活动图的形式化基础:自由选择工作流网

本节首先定义转换的源模型即活动图,并给出一个包含交叠模式的流程模型;然后介绍活动图的形式化表示,即自由选择工作流网;最后定义活动图到其形式化表示的映射方法.

2.1 源模型活动图的定义

一个活动图 $AD=(Nodes,Edges)$ 包括节点集合 $Nodes$ 和边集合 $Edges$.节点集合 $Nodes=EN\cup\{in,fin\}\cup BN\cup CN$. EN 是可执行节点集合,代表流程中的可执行活动. in 和 fin 分别代表流程唯一的开始与结束节点. BN 和 CN 是控制节点: BN 代表分支控制,包含分支选择节点集合 DN 和分支汇合节点集合 MN ; CN 代表并发控制,包含并发分叉节点集合 JN 和并发汇合节点集合 FN .边集 $Edges\subset Nodes\times Nodes$ 是控制流的集合.另外, $Guard:Edges\cap (DN\times Nodes)\rightarrow conditions$ 定义了 DN 输出边上的分支条件,规定一个 DN 所有输出边上的条件集合是互斥和无遗漏的.

图 1 是一个网上购物生成订单流程的活动图模型.流程开始时,根据用户付款方式选择不同的分支:若是在线支付,在付款完毕后开始一个并发过程:由用户确认地址,然后确认订单,同时,系统在奖励用户相应积分后,记录用户的购物偏好,流程完毕;如果选择线下支付,也有一个并发过程开始,在记录用户偏好的同时,由用户选择提货方式,若选择送货上门,则扣除相应积分作为担保,再确认地址,否则需要选择自提的网点.然后确认订单,流程完毕.这个活动图包含交叠模式,它的每次执行都能正确完成,是一个正确的流程模型.

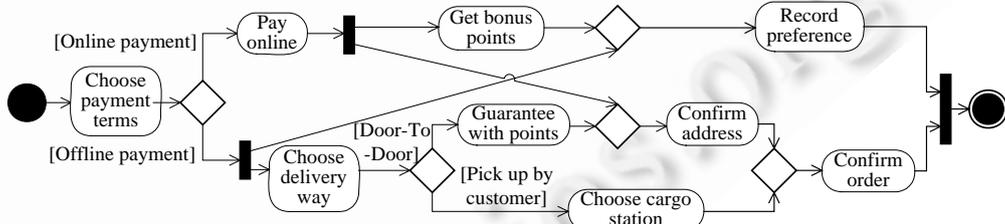


Fig.1 Activity diagram of an order creation process in online purchase

图 1 一个网上购物生成订单流程的活动图

对图 1 中的流程模型,若将其映射为 BPEL 中含有 control links 的 Flow 活动,则生成的代码将包含多达 19 个 links,缺乏可读性.而且由于 links 不能构成环路,此方法对包含循环的流程完全失效.再考虑传统的处理方式“复制相关活动”,由于这种方法不存在形式化的描述,在面对结构复杂的交叠模式时,直观转换方法难免出错,更缺乏转换等价性的形式化证明.另外,并没有研究工作讨论它的适应范围.本文将以此活动图为例,说明对交叠模式的分析与转换方法.

2.2 自由选择工作流网

工作流网是主流的形式化流程定义模型之一,它抽离于数据、层次、时间等扩展定义了流程的控制流结构. Petri 网 $N=(P,T,F)$ 是工作流网,当且仅当各有一个源库所 $i \in P$ 和一个漏库所 $o \in P$,使得 $i=0, o=0$,且任意节点 $x \in P \cup T$ 都在从 i 到 o 的路径上.称 $\bar{N}=(\bar{P},\bar{T},\bar{F})$ 为 N 的短路网,其中, $\bar{P}=P, \bar{T}=T \cup \{t^*\}, \bar{F}=F \cup \{(o,t^*), (t^*,i)\}$.称 (\bar{N},i) 为 N 的短路网系统, t^* 为短路变迁.合理性(soundness)是对工作流网行为属性的规定,要求网在初始标识 $M_0=i$ 下的每一次执行都以正确的方式完成,并且不存在死变迁.判断一个工作流网是否合理,可以通过判断它的短路网系统的活性与有界性进行.关于工作流网更详细的内容,请参阅文献[2].

为了给出活动图的形式化表示,进一步将工作流网限制在自由选择(free choice,简称 FC)网中.FC 网是一类重要的 Petri 网,它通过限制选择与并发间的相互作用,在建模能力与可分析能力间取得良好的折中.Petri 网 $PN=(P,T,F)$ 是自由选择网,当且仅当对于任意两个变迁 $t_1, t_2 \in T$,如果 $\cdot t_1 \cap \cdot t_2 \neq \emptyset$,则有 $|\cdot t_1|=|\cdot t_2|=1$.满足自由选择属性的工作流网就是 FC 工作流网,将合理的 FC 工作流网用 SFCW(sound FC WF-net)网表示.自由选择性在工作流网的建模中被认为是一种应尽量满足的特性^[2]:首先,冲突与并发混合造成的混惑通常造成一个 Petri 网不满足 FC 定义,而这种结构是造成流程模型异常行为的潜在源,在建模中应极力避免;其次,以 FC 网为基础的工作流模型能够被大多数商业流程管理系统支持;最后,借助 FC 网的理论成果能够对流程模型进行更有效和易于理解的分析.基于以上原因,选择自由选择的工作流网作为活动图的形式化基础.

2.3 活动图模型到自由选择工作流网的映射

活动图的形式化表达通过将其映射到自由选择工作流网得到,其中,开始/结束节点及分支控制节点映射到库所,表示可执行活动的节点和并发控制节点映射到变迁;而节点间的控制流映射到相应库所/变迁间的流关系.此外,需添加辅助的节点及流关系,形式化的映射定义见附录. DN 输出边上的条件因抽象建模被忽略.图 1 中的活动图经过映射得到如图 2 所示的工作流网,图中特别标记出所有可执行活动对应的变迁.活动图及映射的定义决定了生成的工作流网都是自由选择的,而一个流程的活动图映射所得的工作流网的合理性表达了对这个流程行为正确性的基本要求.因此,下文只讨论合理的形式化模型,即 SFCW 网.容易证明,图 2 中的工作流网是合理的.

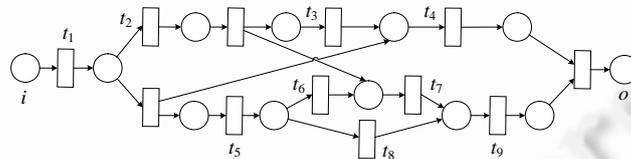


Fig.2 Formal representation of the order creation activity diagram

图 2 生成订单活动图的形式化表达

3 可分离工作流网及到 BPEL 的转换

流程模型中顺序与并发结构能够分别处理的特性有利于图形化流程定义语言向 BPEL 的转换,称这种特性为流程模型的可分离性.本节介绍可分离工作流网的概念,并简述其生成 BPEL 的方法.

3.1 可分离工作流网

只包含顺序结构的 S-网与只包含并发结构的 T-网是定义可分离工作流网的基础.为了给出它的定义,还需描述两种精化操作:对工作流网 N 中一个库所/变迁的精化.具体来说,对库所 s 的精化是用另一个工作流网 N_s 代替 s 在 N 中的位置,如图 3(a)中所示,得到融合 N 与 N_s 的网 N' ;对变迁 t 的精化是用另一个工作流网 N_t 代替 t 在 N 中的位置.注意, N_t 需满足源库所的后集和漏库所的前集都只包含 1 个变迁.如图 3(b)中所示,得到融合 N 与 N_t 的网 N' .两种情况下 N' 仍是工作流网.

定义 1. 满足 S-网定义的工作流网是可分离的;满足 T-网定义且不含有向环路的工作流网是可分离的;用一个可分离工作流网精化另一个可分离工作流网中的变迁/库所所得的工作流网是可分离的.

容易证明,可分离工作流网一定是自由选择的,且满足合理性.图 4 给出一个可分离工作流网的例子.工作流网的可分离性能够通过 SESE(single-enter-single-exit)分解检验,如图 4 所示,虚线框将工作流网分解为相互嵌套或没有交集的多个部分.类似的分解方法在流程模型的分析与转换中得到了广泛应用.其基本思想是,将模型分割为只有 1 条边(或点)作为出/入边(或点)的区域,这些区域间是包含关系或毫无交集,它们是流程分析与转换的基本单元.参照文献[8]中对工作流网的分解方法,将这样的区域定义为工作流网的组件.对于一个可分离工作流网的组件,如果将其直接包含的组件看作抽象的点(库所或变迁,由边界点类型决定),根据可分离工作流网的定义,它一定属于 S-网或无循环 T-网.对图 2 中的工作流网进行 SESE 分解,可知它包含一个混合顺序与并发结构的组件,因此是不可分离的.

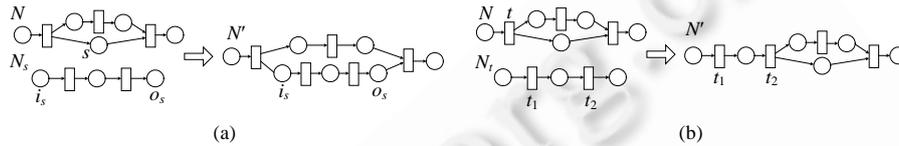


Fig.3 Refinement of a place/transition in a WF-net

图 3 对工作流网库所/变迁的精化

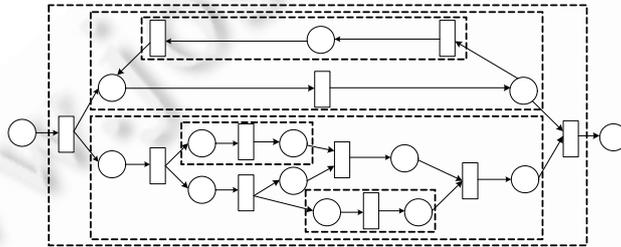


Fig.4 SESE decomposition of a separable WF-net

图 4 可分离工作流网的 SESE 分割

3.2 可分离工作流网生成BPEL

可分离工作流网到 BPEL 的转换以组件为单位由内到外进行.若一个组件只包含 1 个由可执行活动生成的变迁,根据该活动的语义将其转换为 BPEL 的原子活动,如 Invoke,Reply,Receive 和 Assign 等;若它包含子组件,则将其直接包含的子组件看作抽象节点,只关心它直接包含的元素间的结构关系,有利于已有方法对其转换.对属于 S-网的组件,将其与顺序结构化模式匹配,这些模式对应于 BPEL 中的 Sequence,If,While 和 Repeat-Until 等活动.若未找到匹配的模式,则这个组件是非结构化的,可应用已有的非结构化顺序流程模型生成 BPEL 的方法处理它.对属于无循环 T-网的组件,将其与唯一的并发结构化模式(对应于 BPEL 中不含 links 的 Flow 活动)匹配,若无法匹配,则直接映射为包含 links 的 Flow 活动.已有一些方法^[7,10]对这种直接映射进行改进,生成尽可能少的 links,以保护一些结构化的活动,改善代码的可读性.基于以上方法,可分离工作流网到 BPEL 的转换在 SESE 分解的基础上,先从最内部的组件开始生成带有 BPEL 代码的抽象节点,再由内向外逐步转换所有组件,最终生成整个流程的 BPEL 代码.

4 含有交叠模式的 SFCW 网的转换

本节分析交叠模式的形成原因,在此基础上给出含有交叠模式的 SFCW 网到可分离工作流网的转换方法,并证明转换的等价性.为了建立源模型到 BPEL 的映射,要将自由选择工作流网转换成可分离工作流网.对一个

SFCW 网进行 SESE 分解,若它是可分离的,则无须转换;否则,将网中不可分离的结构称为交叠模式(overlapped pattern,简称 OP).通过 SESE 分解,OP 被限制在包含它的最小组件内,不影响其余可分离部分的转换,因此可以只关注包含交叠模式的组件.根据组件的特性,可将其看作一个含有 OP 且不含任何子组件的 SFCW 网.下文中考虑的转换对象都不再包含子组件.

4.1 LBFC 短路网系统的合成

本节介绍的、有界的自由选择(live and bounded FC,简称 LBFC)网系统的合成规则以及这些规则在 SFCW 网中的应用.它们是分析交叠模式的基础.令 N 是 SFCW 网, i 是源库所,则短路网系统 (\bar{N}, i) 一定是 LBFC 的,称为 LBFC 短路网系统.Esparza 在文献[17,18]中研究了 LBFC 网系统的化简与合成规则,为这类系统的结构与属性验证提供了行之有效的方法.本文基于 SFCW 网 N 与它的 LBFC 短路网系统 (\bar{N}, i) 间的一一对应关系,应用 LBFC 网系统的合成规则分析 (\bar{N}, i) 的结构,从而分析 N 的结构特征.为了使这些规则能够应用于 SFCW 网,需要对其增加限制条件.

一条合成规则规定了一个比较简单的网系统如何生成比较复杂(节点和流关系增加)的网系统,此过程中特定属性被保持.LBFC 网系统的合成规则保持的属性包括结构属性(自由选择性和行为属性(活性和有界性)).所有规则按应用范围分为两组:本地规则 S_1, S_2 ;非本地规则 S_3, S_4 .本地规则是指应用条件及对网系统的影响只涉及网中待合成的节点及其邻近结构^[17].图 5 给出 S_1 和 S_2 的图形化描述: S_1 是对一个本地状态的分割,目标库所 p_0 被分割为两个新库所, p_0 中的托肯分散到新库所中去; S_2 是对一个动作的分解,目标变迁 t_0 被分解为两个新变迁.

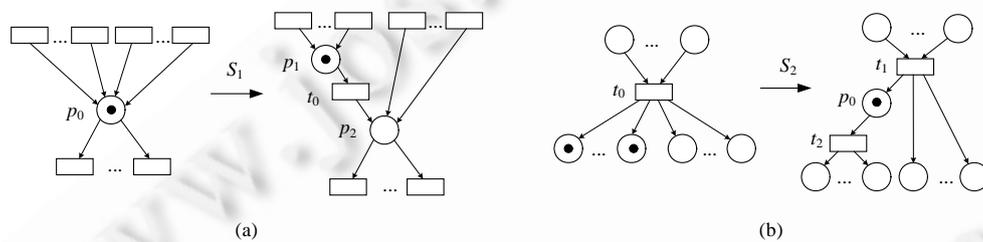


Fig.5 Application of the synthesis rule S_1 and S_2

图 5 合成规则 S_1 和 S_2 的应用

非本地规则 $S_3(S_4)$ 是在网系统中添加一个非负线性相关库所(变迁)及其流关系的操作,因此需要引入非负线性相关库所(变迁)的概念.Petri 网 $N=(P, T, F), |P|>1(|T|>1), C$ 是 N 的关联矩阵, $p \in P(t \in T)$ 是非负线性相关库所(变迁),当且仅当 $p(t)$ 在 C 中的行(列)向量 $r(p)(l(t))$ 是 C 中其余所有行(列)向量的非负线性组合.由于非负线性相关库所和变迁的定义可能涉及整个网的结构,因此 S_3 和 S_4 在应用时需关注全局.为了保持自由选择性,规则 S_3 要求新增库所 p 满足 $|p|=1$,而 S_4 要求新增变迁 t 满足 $|t|=1$.此外,规则 S_3 可能根据需要在 p 中添加托肯以保持系统活性.所有规则的详细内容请参考文献[17,18].

将 LBFC 网系统的合成规则应用于一个 LBFC 短路网系统时,系统的活性、有界性及自由选择性被保持.为了分析 SFCW 网,还需要保护短路网系统的结构特征,否则,合成得到的 LBFC 网系统可能不再是短路网系统,也就不存在对应的 SFCW 网,因此需要对合成规则添加限制条件.能够证明 S_1 和 S_2 应用于一个 LBFC 短路网系统时保持其结构特征,而 S_3 和 S_4 需添加额外的应用条件.

定理 1. (\bar{N}, i) 是一个 LBFC 短路网系统,其中, $N=(P, T, F)$ 是 SFCW 网, i, o 分别是源库所和漏库所, t^* 是短路变迁.在 (\bar{N}, i) 上使用合成规则 $S_k(k=1,2,3,4)$,得到 LBFC 网系统 (N', M'_0) :

- (1) $k=1,2$ 时, (N', M'_0) 仍是短路网系统;
- (2) $k=3$ 时,令添加的库所为 p' ,且有 $r(p') = \sum c_i r(p_i)(c_i > 0, p_i \in P')$,其中, $P' \subset P$.当 i 和 o 都不属于 P' 时, (N', M'_0) 仍是短路网系统;
- (3) $k=4$ 时,令添加的变迁为 t' ,且有 $l(t') = \sum c_i l(t_i)(c_i > 0, t_i \in T')$,其中, $T' \subset T \cup \{t^*\}$.当 $t^* \notin T'$ 时, (N', M'_0) 仍是

短路网系统.

证明:

(1) 结合 S_1 与短路网系统的特点可知,当 $k=1$ 时,若 i 或 o 作为目标库所被分割,则总能在新生成的库所中挑选一个作为 N' 的源/漏库所,若未被分割则继续担任原来的角色,且初始标识总能满足只有源库所中包含一个托肯;当 $k=2$ 时,若 t^* 作为目标变迁被分割,则选择新生成的库所为漏库所,否则, i, o, t^* 继续担任原来的角色.此过程中无托肯的增减,因此 $M'_0 = i$. 综上,经过 S_1 和 S_2 转换, (N', M'_0) 仍是短路网系统.

(2) 在 (\bar{N}, i) 中添加库所 p' 时,由于 i 和 o 的行向量不参与 $r(p')$ 的构成,不会生成 t^* 上的流关系,短路网的结构被保持.为了使 $M'_0 = i$,需满足 p' 中无须添加托肯.根据 S_3 对初始标识的规定, p' 中不加托肯的条件是 N' 的任意非负 S_- 不变量**的支撑集都至少有 1 个库所含有托肯^[18],即都包含源库所 i .LBFC 网系统 (\bar{N}, i) 一定满足这个属性,因此只需考虑 N' 中支撑集包含 p' 的非负 S_- 不变量.令 I' 是这样的不变量,将其支撑集中的 p' 替换为库所集合 P' ,则会构成 (\bar{N}, i) 的一个非负 S_- 不变量 I ,由于 I 的支撑集一定包含 i ,而 $i \notin P'$,因此 I' 的支撑集一定含有 i (详细证明过程见附录).因此, (N', M'_0) 仍是短路网系统.

(3) 在 (\bar{N}, i) 中添加变迁 t' 时,由于 t^* 的列向量不参与 $l(t')$ 的构成,不会生成以 i 为目标或以 o 为源的流关系,短路网的结构被保持.添加变迁的操作不会引起初始标识的变化,则 $M'_0 = i$. 因此, (N', M'_0) 仍是短路网系统. \square

观察 S_1 和 S_2 的特征,它们都只对本地结构造成影响,不会生成顺序与并发交织的结构;而 $S_3(S_4)$ 作为全局规则,在不影响顺序(并发)结构的前提下,通过加入一个库所(变迁)在 FC 网中引入并发(顺序),因此合成过程中可能生成 OP.由于 SFCW 网与它的 LBFC 短路网系统间是一一对应关系,前者包含 OP 当且仅当后者也包含,再根据定理 1 中对 S_3 和 S_4 的限制,在合成过程中,源库所、漏库所及短路变迁都无变化,因此只考虑 S_3 和 S_4 在对应的工作流网上引起的结构改变,即以 SFCW 网为合成规则的应用对象.

4.2 S_3 型交叠模式分析及转化

本节介绍由合成规则 S_3 引起的交叠模式,分析其形成过程,并给出相应的转换方法,最后证明等价性.拥有 SESE 特性的组件^[8]是分析可分离工作流网的重要结构,为了分析不可分离的结构,在讨论 S_3 型交叠模式之前,先扩展组件的定义,将其出/入边界由唯一的节点扩展到节点集合.

定义 2. 工作流网 $N=(P, T, F), C$ 是 N 的一个扩展组件当且仅当 $C \subseteq P \cup T$, 并且存在两个集合 $I_c, O_c \subseteq C$, 满足 $I_c \cup O_c = \emptyset, (C \setminus I_c) \subseteq C \setminus O_c, (C \setminus O_c) \subseteq C \setminus I_c, \forall x, y \in I_c \cup O_c, (x, y) \notin F, I_c(O_c)$ 称为 C 的入(出)边界集合.

4.2.1 S_3 型交叠模式

为了便于讨论,考虑在只有顺序结构,即满足 S -网定义的工作流网中通过 S_3 引入并发的情况.上一节中提到,新增库所 p' 只有 1 个后继变迁,因此当 p' 只有 1 个前驱变迁时,结合 S -网的特性,并发能够分隔到新生成的组件中,不会生成 OP;当 p' 有多个前驱变迁时,由于没有新组件生成,使得顺序与并发无法通过 SESE 分割开来,即形成 S_3 型 OP.下面详细讨论这两种情况.

在一个满足 S -网定义的工作流网上应用 S_3 ,得到工作流网 $N=(P, T, F)$,由定理 1 可知, N 一定是 SFCW 网.令 p' 是新加入的库所,它的行向量 $r(p') = \sum c_i r(p_i) (c_i > 0, p_i \in P'), P' \subset P \wedge \{i, o\} \cap P' = \emptyset$. 若将 P' 中的库所看作一个整体,它们构成与 p' 拥有相同前/后集的结构;而这两者结合,形成 p' 与 P' 中所有库所是并发关系的新结构.形式化描述这一过程:引入 p' 前, P' 及其前后集 $P' \cup P'$ 形成一个只包含顺序结构的扩展组件 C ,而 $C' = C \cup \{p'\}$ 是引入 p' 后与原来的扩展组件 C 有相同出/入边界集合的包含并发与顺序的扩展组件,它的入边界是 p' 的前集,出边界是 p' 唯一的后继变迁.

当 p' 只有 1 个前驱变迁时, C' 是满足 SESE 特性的组件.尽管其中包含并发,因为并发只在它的出/入边界点上发生,而作为入边界点的变迁只有 1 个后继库所,同样作为出边界点的变迁只有 1 个前驱库所,以这两个库所

** 向量 I 是 Petri 网 $N=(P, T, F)$ 的非负 S_- 不变量,当且仅当 $C^T \cdot I = \theta, I \geq 0$, 其中, C^T 是 N 的邻接矩阵的转置. $P_1 = \{p \in P | I(p) \neq 0\}$ 称为 I 的支撑集.

为边界形成一个新的组件,使 C' 中的并发能够通过 SESE 分解与顺序结构分隔开来,所以它仍是可分离的.

当 p' 有多个前驱变迁时,原扩展组件 C 的“接口”类似于活动图中的合并节点,称其为“类合并”的扩展组件.在新构成的扩展组件 C' 中, p' 的加入不会生成任何新组件.此时, C' 中的顺序和并发结构无法通过 SESE 分解而分离,因此生成一个 OP,称这样的 OP 为 S_3 型 OP.下面给出其形式化定义.

定义 3. 在一个 SFCW 网 N 中, p' 是非负线性相关库所, $r(p') = \sum c_i r(p_i) (c_i > 0, p_i \in P'), |\cdot p'| > 1, C = P' \cup T'$ 是引入 p' 前的一个只包含顺序结构的“类合并”的扩展组件, $T' = P' \cup P'$, 由扩展组件 $C' = C \cup \{p'\}$ 及其节点间的流关系构成的子网称为 S_3 型 OP.

以图 6 中的 workflow 网为例, $P' = \{p_i | i=1, \dots, 5\}$, P' 和变迁集合 $\{t_i | i=5, \dots, 12\}$ 形成只包含顺序结构的“类合并”的扩展组件 C , 引入非负线性相关库所 p' 后, 得到扩展组件 C' , 由 C' 构成的子网是一个 S_3 型 OP.

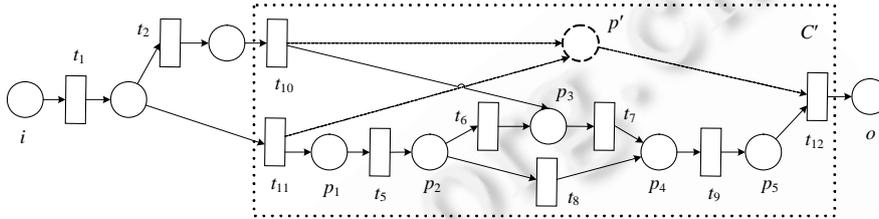


Fig.6 Illustration of the formation of a S_3 -type OP
图 6 一个 S_3 型 OP 的生成示例

一个 S_3 型交叠模式虽然含有并发结构,却没有真正参与并发的活动.考虑到规则 S_1 和 S_4 在添加变迁时不会生成并发结构从而增加分析和转换的复杂性,下面应用 S_1 和 S_4 在一个 S_3 型 OP 中引入新变迁,使流程模型中包含真正参与并发的可执行活动.具体来说,以 OP 中的非负线性相关库所及其前后集构成的子网为对象,使用规则 S_1 和 S_4 对其精化.每应用一次 S_1 , 就有一个库所被分裂为两个局部状态,并有一个新变迁生成;每应用一次 S_4 , 就有一个新变迁加入,并在子网中形成无向环路.将一个 S_3 型 OP 经上述合成过程生成的结构称为 $S_1 \& S_4$ 精化的 S_3 型交叠模式.下面给出其形式化定义.

定义 4. N_{OP} 是 workflow 网 N 中的一个 S_3 型 OP, 线性相关库所 p' 与其前集 $\{t_1, t_2, \dots, t_k\}$ 、后集 $\{t_0\}$ 构成子网 $N_1 = (\{p'\}, \{t_1, t_2, \dots, t_k, t_0\}, \{(t_i, p') | 1 \leq i \leq k\} \cup \{p', t_0\})$, 以 N_1 为对象, 若干次应用规则 S_1 和 S_4 , 称转换后的 N_{OP} 为 $S_1 \& S_4$ 精化的 S_3 型 OP.

以图 6 中的 S_3 型 OP 为例, 其中 $N_1 = (\{p'\}, \{t_{10}, t_{11}, t_{12}\}, \{(t_{10}, p'), (t_{11}, p'), (p', t_{12})\})$, 在 N_1 上两次应用 S_1 , 分别引入变迁 t_3 和 t_4 , 得到最右边的子网 N_2 , 如图 7 所示. 图 6 中的 workflow 网经上述合成过程生成的正是图 2 中的 workflow 网, 它包含一个 $S_1 \& S_4$ 精化的 S_3 型 OP, 其中, $\{t_3, t_4\}$ 与 $\{t_i | i=5, \dots, 9\}$ 对应的可执行活动都可能参与并发. 下文的转换以 $S_1 \& S_4$ 精化的 S_3 型 OP 为对象, 并简称其为 S_3 型 OP.

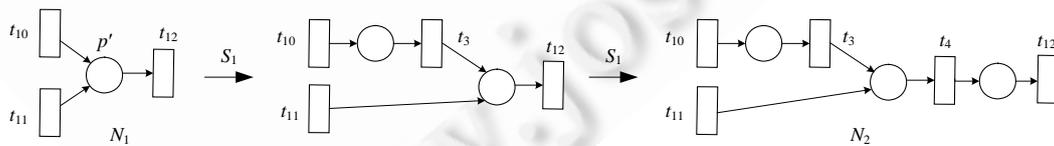


Fig.7 Application of the rule S_1 to a S_3 -type OP
图 7 规则 S_1 在 S_3 型 OP 上的应用

4.2.2 S_3 型交叠模式转换方法

含有 S_3 型 OP 的 SFCW 网到可分离 workflow 网的转换, 利用网展开的概念将 OP 中每条可选分支分别展开, 在此过程中, 相关变迁被复制多次, 分支的汇合向后推延, 这是对“复制相关活动”方法的形式化描述. 而 S_3 型 OP 只有 1 个变迁作为出边界的特点, 使得对每条分支的展开都能得到一个新的包含并发的可分离组件, 且所有分

支最后都汇合在一个库所上.通过展开分支及推迟分支的汇合,将并发隔离在新组件中,达到分离顺序与并发结构的目的.具体地,设 N_{OP} 是 workflow 网 N 中的 S_3 型 OP,它是分别以 $\{t_1, t_2, \dots, t_k\}$ 和 $\{t_0\}$ 为出/入边界的扩展组件构成的子网,分别以每个入边界点 t_i 为起点展开 N_{OP} ,到唯一的出边界点 t_0 终止,即删除 N_{OP} 中所有不在 t_i 到 t_0 路径上的节点及流关系,构造以 t_i 和 t_0 为出/入边界的组件.最后,所有组件在 t_0 的后继库所处汇聚, N 中其余结构保持不变.下面给出转换的形式化表达,并证明所得的 workflow 网一定是可分离的.

$N=(P,T,F)$ 是只包含 1 个 S_3 型 OP: $N_{OP}=(P_1, T_1, F_1)$ 的 SFCW 网, N_{OP} 是分别以 $\{t_1, t_2, \dots, t_k\}$ 和 $\{t_0\}$ 为出/入边界的扩展组件构成的子网.首先,分别构造组件 $C_i(1 \leq i \leq k): C_i = \{x^i | x \text{ 在 } t_i \text{ 到 } t_0 \text{ 的路径上}\}$,其中, x^i 称为节点 x 在 C_i 中的副本,特别地,只在 1 个 C_i 中存在副本的 t_i 与 t_i^i 不作区分.转换得到可分离 workflow 网 $N'=(P', T', F')$:

- 1) $P' = P \setminus P_1 \cup \bigcup_{i=1}^k \{p | p \in C_i\}$;
- 2) $T' = T \setminus T_1 \cup \bigcup_{i=1}^k \{t | t \in C_i\}$;
- 3) $F' = F |_{P_1 \cup T_2} \cup \bigcup_{i=1}^k \{(x^i, y^i) | x^i, y^i \in C_i, (x, y) \in F\} \cup \bigcup_{i=1}^k \{(t_i^i, p_0)\}$, 其中, $F|_X$ 表示在 F 中删除所有包含集合 X 中节点的流关系, $T_2 = T_1 \setminus \{t_i | 1 \leq i \leq k\}$, p_0 是 t_0 在 N 中的后继库所.

在一个 S_3 型 OP 中,原来的“类合并”扩展组件只包含顺序结构,而引入 S_3 型 OP 再经 S_1 与 S_4 精化后所得子网也只包含顺序结构,这两个结构在针对每个入边界点展开的组件中也都只包含顺序结构,且两部分的节点间没有任何流关系,并发只在组件的出/入边界点上发生,因此生成的 workflow 网是可分离的.形式化证明如下:

定理 2. 包含 S_3 型 OP 的 SFCW 网 N 经上述转换生成的网 N' 一定是可分离 workflow 网.

证明:令子网 N_{OP} 为所含 OP,引入 S_3 型 OP 的库所 p^i 及其前集 $\{t_1, t_2, \dots, t_k\}$,后集 $\{t_0\}$ 构成的子网再经 S_1 与 S_4 精化所得子网为 N_2 ,则 N_{OP} 可看作是 N_2 与 N_{OP} 中的“类合并”扩展组件 C 之间的并发,除公共节点集合 $\{t_1, t_2, \dots, t_k\}$ 和 $\{t_0\}$ 外,分别来自 N_2 与 C 的节点之间不存在任何流关系.在针对 t_i 构造的组件 C_i 中,考虑 t_i 的后集中那个属于 C 的库所 p_1^i 及 t_0^i 的前集中那个属于 C 的库所 p_2^i ,这两个库所间的所有节点都在 p_1^i 到 p_2^i 的路径上,形成一个以 p_1^i 和 p_2^i 为边界的满足 S-网定义的组件 C_i' .类似地, N_2 在 C_i 中的副本也形成一个满足 S-网定义的组件 C_i'' ,因此在 C_i 中形成 C_i' 与 C_i'' 并发的结构,且并发在 t_i 处开始, t_0^i 处聚合,所以 C_i 是可分离的.此外, N' 中再无并发结构,因此它是可分离的. □

对图 2 中的 workflow 网按上述过程进行转换,得到如图 8 所示的可分离 workflow 网. C_1 和 C_2 是分别针对变迁 t_{10} 和 t_{11} 生成的组件,它们各自以 t_{12} 的两个副本 t_{12}^1 和 t_{12}^2 为出边界点,而 t_{12}^1 和 t_{12}^2 都指向 t_{12} 原来的后继库所.从图中可看出,转换所得的可分离 workflow 网与原来的流程模型相比包含更多的结构化活动,因此能够改进生成的 BPEL 代码的可读性.鉴于图 8 中的 workflow 网到 BPEL 的转换比较直观,本文不再给出相应的 BPEL 代码.

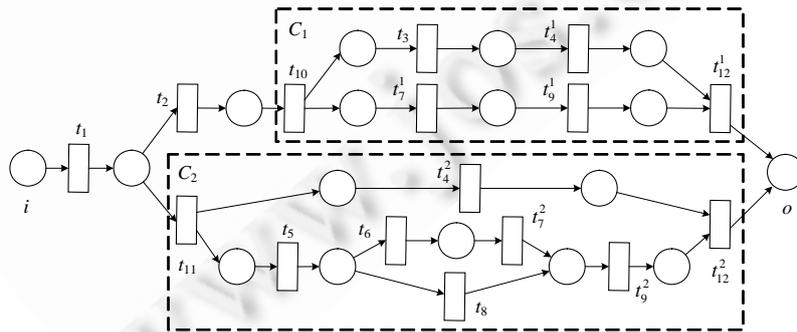


Fig.8 Separable WF-net derived from the WF-net in Fig.2

图 8 图 2 中的 workflow 网转换所得的可分离 workflow 网

4.2.3 S_3 型交叠模式转换的等价性证明

本节利用并发正则表达式(concurrent regular expression,简称 CRE)^[19]的概念证明转换的等价性. CRE 是与 Petri 网语言等价的表达式,它提供了一种 Petri 网语言的模块化表达.文献[20]介绍了 S -网的 CRE 描述方法,并证明一个结构复杂 Petri 网的 CRE 表达式可通过一组 S -网的同步合成运算得到.基于文献[19,20]中的理论,证明转换前后的两个工作流网系统等价的方法如下:首先,分别将两个网系统分解为两个 S -网,再证明分解所得 S -网的 CRE 表达式两两等价,可知两个工作流网系统的 CRE 表达式等价,推出两个工作流网等价.需要强调的是,等价性证明并不特定于某两个转换前后的工作流网,而是对所有能够应用上一节中转换方法的工作流网都成立.

定理 3. 包含 S_3 型 OP 的 SFCW 网 N 与转换后的可分离工作流网 N' 等价.

证明:分别分解 SFCW 网 N 与可分离工作流网 N' .首先考虑 N ,称 N 中使用规则 S_3 引入 OP 前的子网为 N_1 ,而引入 S_3 型 OP 的库所及其前后集构成的子网再经 S_1 & S_4 精化所得的子网为 N_2 ,则 N 可分解为两个 S -网 N_1 和 N_2 .再考虑 N' ,将 N_2 在 N' 中的所有副本组成的 S -网 N'_2 从 N' 中分离出来,其余的结构构成子网 N'_1 ,则 N' 可分解为两个 S -网 N'_1 和 N'_2 .然后,分别证明 N_1 与 N'_1 、 N_2 与 N'_2 的 CRE 表达式相等.首先考虑 N_1 与 N'_1 , N_1 中包含引入 S_3 型 OP 前的“类合并”扩展组件 C ,而 N'_1 中包含 C 的若干副本,它们共同组成与 C 有相同出入边界的扩展组件 C' ,它可看作是对 C 的部分展开.除顺序结构 C 与 C' 外, N_1 与 N'_1 拥有完全相同的顺序结构,因此两者的 CRE 表达式相等.对于 N_2 与 N'_2 ,由于 N'_2 是对 N_2 中所有路径的展开,它们的 CRE 表达式都等于这些路径的同步合成,因此两者的 CRE 表达式相等.最后,根据一个 Petri 网的 CRE 表达式等于它分解所得 S -网的 CRE 表达式的同步合成可知, N 与 N' 的 CRE 表达式相等,即两个工作流网等价.详细证明过程见附录. \square

以图 2 中包含 S_3 型 OP 的 SFCW 网 N 与图 8 中转换所得的可分离工作流网 N' 为例说明等价性证明过程.首先将 N 分解为 N_1 和 N_2 :由图 6 中的工作流网移除 p' 及其流关系可得 N_1, N_2 即图 7 中的 N_2 . N' 分解为 N'_1 和 N'_2 ,如图 9 所示.其中, N'_1 包含对 N_1 中“类合并”扩展组件的展开,两者可看作接受相同正则语言的自动机. N'_2 将 N_2 展开为两条路径,两者的 CRE 表达式都等于这两条路径的同步合成,因此 N 与 N' 等价.

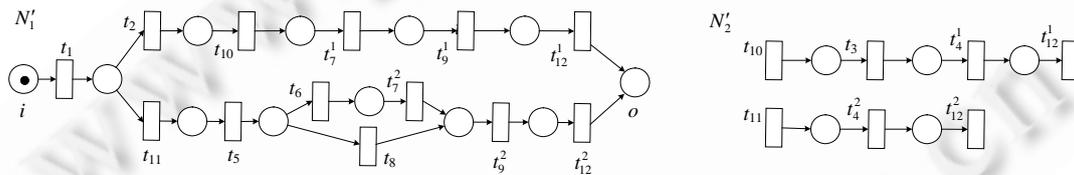


Fig.9 S-Nets derived from the decomposition of N'

图 9 对网 N' 分解所得的 S -网

至此,包含一个 S_3 型 OP 工作流网的转换方法及等价性证明介绍完毕,它能够推广到包含多个互不相交的 S_3 型 OP 的工作流网上.具体地,在一个只包含顺序结构的工作流网 N 上,多次应用规则 S_3 引入 m 个非负线性相关库所 $p'_i (1 \leq i \leq m)$,再分别应用 S_1 和 S_4 生成 S_1 & S_4 精化的 S_3 型 OP,且保证每个 OP 都互不相交.对每个 OP 都按照上述方法进行转换,生成的工作流网仍是可分离的.若将 N 分解为多个 S -网,则包括一个只包含顺序结构的工作流网及 m 个只包含顺序结构的子网,且这 m 个子网间没有交集.因此,转换等价性也能由包含一个 S_3 型 OP 的情况推广得到.如果有任意两个 OP 存在交集,则以上方法不再适用.然而在实际应用中,这样的流程设计非常少见.

4.3 S_4 型交叠模式

与 S_3 型 OP 类似,通过规则 S_4 在只包含并发的结构中引入顺序而形成的不可分离结构就是 S_4 型 OP.下面简要分析 S_4 型 OP 的生成过程.向一个满足 T -网定义且不含有向环路的工作流网 $N=(P, T, F)$ 中加入一个非负线性相关变迁 t', t' 的列向量 $l(t') = \sum c_i l(t_i) (c_i > 0, t_i \in T'), T' \subset T$. T' 加上它的前后集形成一个引入 t' 前的扩展组件 C ,引入 t' 后, C 与 t' 构成一个扩展组件 C' ,分别以 t' 的前集和后集作为出边界和入边界,且包含顺序与并发结构.由于

t' 只能有 1 个前驱库所,当它只有 1 个后继库所时, C' 是可分离的组件;若 t' 的后继库所个数大于 1, C' 构成的子网中包含顺序与并发无法分离的结构,则生成一个 S_4 型 OP.再对 t' 及其前后集形成的子网应用规则 S_2 与 S_3 ,形成 S_2 & S_3 精化的 S_4 型 OP.

对含有一个 S_4 型 OP 的 SFCW 网,上节中复制相关活动的方法能够处理特定的情况,要求网中存在一个扩展组件 C_1 ,它的入边界集合是引入这个 OP 的变迁 t' 的后集,出边界只有 1 个变迁.转换方法是分别针对 t' 及引入 t' 前的扩展组件 C 展开 t' 唯一前驱库所上的两条选择路径, C_1 被复制两次,而 C_1 只有 1 个变迁作为出边界的特点使得两次展开都能形成可分离的组件,从而得到可分离的工作流网.转换等价性仍可利用 CRE 表达式相等证明,这里不再赘述.

5 结束语

本文借助工作流网形式化地分析了活动图中的交叠模式,按照 LBFC 网系统中引入并发的合成规则 S_3 及引入顺序的合成规则 S_4 定义了两类交叠模式: S_3 型及 S_4 型 OP,给出 S_3 型 OP 转换方法的形式化描述及等价性证明,并简述部分 S_4 型 OP 的转换方法,由此总结了“复制相应活动”方法的适用范围.通过两次映射、一次转换完成活动图生成 BPEL 的过程,以形式化模型为依托,保证了转换的可靠性.研究还有很多不足之处,例如,在规则 S_3 和 S_4 上添加限制条件时,仅证明了限制条件保持短路网结构特征的充分性.另外,对于不能使用复制活动转换的部分 S_4 型 OP,还需探索新的方法.

References:

- [1] OASIS: Web services business process execution language version 2.0. 2007. <http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html>
- [2] van der Aalst WMP. Workflow verification: finding control-flow errors using Petri-net-based techniques. In: van der Aalst WMP, Desel J, Oberweis A, eds. Proc. of the Business Process Management: Models, Techniques, and Empirical Studies 2000. LNCS 1806, Heidelberg: Springer-Verlag, 2000. 161–183.
- [3] Hauser R, Koehler J. Compiling process graphs into executable code. In: Karsai G, Visser E, eds. Proc. of the Int'l Conf. on Generative Programming and Component Engineering 2004. LNCS 3286, Heidelberg: Springer-Verlag, 2004. 317–336. [doi: 10.1007/978-3-540-30175-2_17]
- [4] Koehler J, Hauser R, Sendall S, Wahler M. Declarative techniques for model-driven business process integration. IBM Systems Journal, 2005,44(1):47–65. [doi: 10.1147/sj.441.0047]
- [5] Koehler J, Hauser R. Untangling unstructured cyclic flows—A solution based on continuations. In: Meersman R, Tari Z, eds. Proc. of on the Move to Meaningful Internet Systems 2004: CoopIS, DOA, and ODBASE. LNCS 3290, Heidelberg: Springer-Verlag, 2004. 121–138. [doi: 10.1007/978-3-540-30468-5_10]
- [6] Zhao W, Hauser R, Bhattacharya K, Bryant BR, Cao F. Compiling business processes: Untangling unstructured loops in irreducible flow graphs. Int'l Journal on Web Grid Services, 2006,2(1):68–91. [doi: 10.1504/IJWGS.2006.008880]
- [7] Zhao W, Bryant BR, Cao F, Hauser R, Bhattacharya K, Tao T. Transforming business process models in the presence of irreducibility and concurrency. Int'l Journal on Business Process Integration and Management, 2007,2(1):37–48. [doi: 10.1504/IJBPI.2007.014103]
- [8] van der Aalst WMP, Lassen KB. Translating unstructured workflow processes to readable BPEL: Theory and implementation. Information and Software Technology, 2008,50(3):131–159. [doi: 10.1016/j.infsof.2006.11.004]
- [9] Ouyang C, Dumas M, Breutel S, ter Hofstede AHM. Translating standard process models to BPEL. In: Dubois E, Pohl K, eds. Proc. of the Advanced Information Systems Engineering 2006. LNCS 4001, Heidelberg: Springer-Verlag, 2006. 417–432. [doi: 10.1007/11767138_28]
- [10] Ouyang C, Dumas M, van der Aalst WMP, ter Hofstede AHM, Mendling J. From business process models to process-oriented software systems. ACM Trans. on Software Engineering and Methodology, 2009,19(1):1–37. [doi: 10.1145/1555392.1555395]

- [11] Hauser RF, Friess M, Küster JM, Vanhatalo J. An incremental approach to the analysis and transformation of workflows using region trees. *IEEE Trans. on Systems, Man, and Cybernetics—Part C: Applications and Reviews*, 2008,38(3):347–359. [doi: 10.1109/TSMCC.2008.919178]
- [12] Hauser R. Analysis and transformation of behavioral models containing overlapped patterns. *Journal of Object Technology*, 2010, 9(3):105–124. [doi: 10.5381/jot.2010.9.3.a4]
- [13] Li JQ, Fan YS, Zhou MC. Timing constraint workflow nets for workflow analysis. *IEEE Trans. on System, Man, and Cybernetics—Part A: System and Humans*, 2003,33(2):179–193. [doi: 10.1109/TSMCA.2003.811771]
- [14] Li JQ, Fan YS. A method of workflow model performance analysis. *Chinese Journal of Computers*, 2003,26(5):513–523 (in Chinese with English abstract).
- [15] Ge JD, Hu H, Lü J. A transformation approach from workflow net to PERT diagram based on invariants. *Acta Electronica Sinica*, 2008,36(5):893–898 (in Chinese with English abstract).
- [16] Pang SC, Jiang CJ. Workflow performance analysis based on invariant decomposition algorithm. *Chinese Journal of Computers*, 2010,33(5):908–918 (in Chinese with English abstract). [doi: 10.3724/SP.J.1016.2010.00908]
- [17] Esparza J. Reduction and synthesis of live and bounded free choice Petri nets. *Information and Computation*, 1994,114(1):50–87. [doi: 10.1006/inco.1994.1080]
- [18] Esparza J, Silva M. Top-Down synthesis of live and bounded free choice nets. In: Rozenberg G, ed. *Proc. of the Advances in Petri nets'91*. LNCS 524, Heidelberg: Springer-Verlag, 1991. 118–139. [doi: 10.1007/BFb0019972]
- [19] Garg VK, Ragunath MT. Concurrent regular expressions and their relationship to Petri nets. *Theoretical Computer Science*, 1992, 96(2):285–304. [doi: 10.1016/0304-3975(92)90339-H]
- [20] Zeng QT. Behavior descriptions of structure-complex Petri nets based on synchronous composition. *Journal of Software*, 2004, 15(3):327–337 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/15/327.htm>

附中文参考文献:

- [14] 李建强,范玉顺.一种 workflow 模型的性能分析方法. *计算机学报*,2003,26(5):513–523.
- [15] 葛季栋,胡昊,吕建.一种基于不变量的从 workflow 网到 PERT 图的转换方法. *电子学报*,2008,36(5):893–898.
- [16] 庞善臣,蒋昌俊.一种基于不变量结构分解的 workflow 性能分析方法. *计算机学报*,2010,33(5):908–918.
- [20] 曾庆田.基于同步合成的结构复杂 Petri 网的行为描述. *软件学报*,2004,15(3):327–337. <http://www.jos.org.cn/1000-9825/15/327.htm>

附录

1. 活动图映射到自由选择 workflow 网的形式化定义.

边集 $Edges$ 被划分为 3 个子集:

- $Edges_1 = \{e | e = (s_e, t_e) \wedge \{s_e, t_e\} \subset \{in, fin\} \cup BN\}$;
- $Edges_2 = \{e | e = (s_e, t_e) \wedge \{s_e, t_e\} \subset EN \cup CN\}$;
- $Edges_3 = \{e | e = (s_e, t_e) \wedge \{s_e, t_e\} \cap (\{in, fin\} \cup BN) \neq \emptyset \wedge \{s_e, t_e\} \cap (EN \cup CN) \neq \emptyset\}$.

其中,对于 $\forall e = (s_e, t_e) \in Edges$, s_e 和 t_e 分别表示 e 的源节点和目标节点.在此基础上定义映射:

- 1) $P = \{in, fin\} \cup BN \cup \{p_e | e \in Edges_1\}$;
- 2) $T = EN \cup CN \cup \{t_e | e \in Edges_2\}$;
- 3) $F = \{(s_e, t_e) | (s_e, t_e) \in Edges_3\} \cup \{(s_e, x_e), (x_e, t_e) | e = (s_e, t_e) \in Edges_1 \cup Edges_2\}$.

对于 $\forall e \in Edges_1 \cup Edges_2$, x_e 表示由 e 映射生成的节点.

2. 证明定理 1(2)中的命题“(N', M'_0) 满足 p' 中无须增加托肯的条件”.

证明:令 I 是 (\bar{N}, i) 的任意非负 S_- 不变量,且支撑集不含 i ,则有 $I \cdot M_0 = 0$,这与 (\bar{N}, i) 是活的矛盾.因此, (\bar{N}, i) 中任意一个非负 S_- 不变量的支撑集都包含 i .

由于 (\bar{N}, i) 中的非负 S_- 不变量在 (N', M'_0) 中仍是非负 S_- 不变量,因此只需证明在 (N', M'_0) 中若非负 S_- 不变

量的支撑集含有 p' , 则一定含有 i .

设 I' 是 (N', M'_0) 的非负 S_- 不变量, 支撑集为 $P_1 \cup \{p\} (P_1 \subseteq P)$, 则有 $C' \cdot I' = \theta_{T \cup \{i^*\}}$ (C' 是 N' 的关联矩阵, $\theta_{T \cup \{i^*\}}$ 是分量全为 0 的 T_- 向量), 用 $P_1 \cup \{p\}$ 在 C' 中的行向量表示就是

$$c_0 r(p) + \sum_{p_i \in P_1} c_i r(p_i) = \theta_{T \cup \{i^*\}}^T (c_0 > 0, c_i > 0, i = 1, \dots, |P_1|).$$

已知 $r(p) = \sum_{p_j \in P'} c_j r(p_j)$, 其中 $P' \subseteq P \wedge \{i, o\} \cap P' = \emptyset$, 代入上式中有:

$$c_0 \cdot \sum_{p_j \in P'} c_j r(p_j) + \sum_{p_i \in P_1} c_i r(p_i) = \theta_{T \cup \{i^*\}}^T,$$

则有 $C \cdot I_1 = \theta_{T \cup \{i^*\}}$ (C 是 \bar{N} 的关联矩阵), 其中 I_1 是 \bar{N} 的非负 S_- 不变量, 它的支撑集为 $P_2 = P' \cup P_1$.

已知 $i \notin P'$, 若 $i \notin P_1$, 则有 $i \notin P_2$. 与 (\bar{N}, i) 中任意非负 S_- 不变量的支撑集都包含 i 矛盾, 所以 $i \in P_1$.

所以, (N', M'_0) 中任意非负 S_- 不变量的支撑集一定含有 i , 即满足 p' 中无须增加托肯的条件. □

3. 定理 3“包含 S_3 型 OP 的 SFCW 网 N 与转换后的可分离 workflow 网 N' 等价”的详细证明过程.

证明: 令 $N = (P, T, F)$, 将 P 划分为两个集合: N_2 中的库所集合 P_2 及其余的库所集合 P_1 , 以这个划分构造两个 S -网: $N_1 = (P_1, T_1, F_1), N_2 = (P_2, T_2, F_2)$.

令 $N' = (P', T', F')$, 考虑证明 N' 可分离性(定理 2)时提到的组件 $C_i'' (1 \leq i \leq k)$, 用 P_i'' 表示 C_i'' 上的库所集合, 则 P' 被划分为两个集合: $C_i'' (1 \leq i \leq k)$ 中的所有库所集合 $P_i'' = \bigcup_{i=1}^k P_i''$ 和其余的库所集合 P'_i . 由此构造两个 S -网:

- $N'_1 = (P'_i, T'_i, F'_i)$, 通过在 N' 中删除 $C_i'' (1 \leq i \leq k)$ 得到. 显然, N'_1 是 workflow 网并满足 S -图的定义.
- $N'_2 = \left(P'_2, \bigcup_{i=1}^k (T'_i \cup \{t_i, t'_i\}), \bigcup_{i=1}^k (F'_i \cup \{(t_i, p'_i), (p'_i, t'_i)\}) \right)$, T''_i 和 F''_i 分别是 C_i'' 的变迁集合及其所有节点间的流关系集合, 而 p'_i 和 p''_i 分别是 C_i'' 的出入边界点.

1) N_1 与 N'_1 都是 S -网, 将两个网分别转换为有限自动机, 则自动机接受语言的正则表达式就是它们的 CRE 表达式.

- N_1 对应的自动机 $A_1 = (P_1, \Sigma_1, \delta_1, i, \{o\})$, 状态集是 P_1 , 字母表 $\Sigma_1 = T_1$, 状态转移函数定义为 $\delta_1 = \{(p_1, t, p_2) | t \in T_1, p_1, p_2 \in P_1, t \in p_1 \cdot \wedge t \in \cdot p_2\}$.
- N'_1 对应的自动机 $A'_1 = (P'_i, \Sigma'_i, \delta'_i, i, \{o\})$, 状态集是 P'_i , 字母表 $\Sigma'_i = T_1$, 定义一个标记函数

$$\rho: T'_i \rightarrow T_1, \forall t' \in C_i'', \rho(t') = t; \forall t \notin C_i'', \rho(t) = t.$$

它将 N'_1 的组件 $C_i'' (1 \leq i \leq k)$ 中变迁的副本映射到它们在 N_1 中的原变迁, 实际含义是代表同一动作的若干变迁用同一个字母标记. 而 N_1 和 N'_1 中其余变迁是一一对应的, 所以 ρ 是满射的.

状态转移函数定义为

$$\delta_2 = \{(p_1, \rho(t), p_2) | t \in T'_i, p_1, p_2 \in P'_i, t \in p_1 \cdot \wedge t \in \cdot p_2\}.$$

由于 N_1 和 N'_1 中不经过 $t_i (1 \leq i \leq k)$ 的路径都是一一对应的, 所以只考虑经过 t_i 情况.

令字符串 $w = w_1 \cdot w_2 \cdot w_3$, 其中, w_2 以第 1 个 t_i 开头, 最后一个 t_0 结尾:

① 设 $w \in L(A_1)$, 若 w 只包含 1 个 t_i , 则 t_i 和 t_0 在 w_2 中各只出现 1 次. 在 N'_1 中一定存在一条 t_i 到 t'_0 的路径 w'_2 , 满足 $\rho(w'_2) = w_2$. 则有 $\rho(w_1 \cdot w'_2 \cdot w_3) = w_1 \cdot \rho(w'_2) \cdot w_3 = w$, 所以 $w \in L(A'_1)$. 若 w 包含 n 个 t_i , 则由 N_1 的结构特征可知, $w_2 = w_2^1 \dots w_2^2 \dots w_2^j \dots w_2^n (1 \leq j \leq n)$, 其中, $w_2^j (1 \leq j \leq n)$ 以 t_i 开头、 t_0 结尾, 且 t_i 和 t_0 各只出现 1 次.

类似于只包含 1 个 t_i 的证明方法, 可证 $w \in L(A'_1)$.

② 设 $w \in L(A'_1)$, 若 w 只包含 1 个 t_i , 令 $\rho^{-1}(w_2) = w'_2$, 则 w'_2 是 N'_1 中一条 t_i 到 t'_0 的路径, 且 t_i 和 t'_0 各只出现 1 次, 它对应于 N_1 中一条 t_i 到 t_0 的路径 w_2 , 由此可知, $w \in L(A_1)$.

w 包含多个 t_i 的情况可参考 $w \in L(A_1) \Rightarrow w \in L(A'_1)$ 的证明.

由以上两点可知, $L(A_1) = L(A'_1)$, 即 $L(N_1) = L(N'_1)$.

2) N_2 是包含多个原始变迁(S -网中满足 $|t|=0$ 的变迁)的 S -网,将 N_2 分解为 k 个只包含 1 个原始变迁的 S -网:

$$N_2^i = (P_2, T_2^i, F_2^i), \quad 1 \leq i \leq k,$$

其中,

$$T_2^i = T_2 - \{t_j \mid 1 \leq j \leq k \wedge j \neq i\}, F_2^i = F_2 \cap ((P_2 \times T_2^i) \cup (T_2^i \times P_2)),$$

则 N_2 的 CRE 表达式为

$$L(N_2) = L(N_2^1) \parallel L(N_2^2) \parallel \dots \parallel L(N_2^k).$$

对 N_2^i 进行分解,得到 k 个 S -网 $(N_2^i)'$ ($1 \leq i \leq k$), 其中, $(N_2^i)'$ 由组件 C_i'' 加上 t_i 和 t_0^i 及相应流关系所得,

$$(N_2^i)' = (P_i'', T_i'' \cup \{t_i, t_0^i\}, F_i'' \cup \{(t_i, p_1^i), (p_2^i, t_0^i)\}).$$

N_2^i 的 CRE 表达式为

$$L(N_2^i) = L((N_2^i)') \parallel L((N_2^i)') \parallel \dots \parallel L((N_2^i)').$$

根据 N_2^i 的构造方法,如果将其中不在 t_i 到 t_0 路径上的节点及流关系删除,则它接受的语言不变.再定义一个标记函数将 N_2^i 中同一变迁的若干副本映射到它们在 N_2 中的原变迁上,这样, N_2^i 与 $(N_2^i)'$ 就会接受相同的语言,即 $L(N_2^i) = L((N_2^i)'),$ 所以 $L(N_2) = L(N_2^i).$

3) 由 $L(N) = L(N_1) \parallel L(N_2), L(N') = L(N_1) \parallel L(N_2')$ 及步骤 1)、步骤 2) 推出 $L(N) = L(N').$ □



张曼(1984—),女,陕西西安人,博士生,主要研究领域为商业流程建模,工作流网.



王小兵(1979—),男,博士,副教授,CCF 会员,主要研究领域为 Web 服务建模与验证.



段振华(1948—),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为网络计算,可信软件理论和技术.