

基于 P/T 系统化简方法的工作流过程模型验证*

赵文¹⁺, 袁崇义¹, 刘刚², 张世琨¹, 王立福¹

¹(北京大学 信息科学与技术学院, 北京 100871)

²(解放军信息工程大学, 河南 郑州 450052)

Workflow Process Model Verification Using Reduction Method Based on P/T System

ZHAO Wen¹⁺, YUAN Chong-Yi¹, LIU Gang², ZHANG Shi-Kun¹, WANG Li-Fu¹

¹(Institute of Information Science and Technology, Peking University, Beijing 100871, China)

²(University of PLA Information Engineering, Zhengzhou 450052, China)

+ Corresponding author: Phn: +86-10-62753026, E-mail: owen@cs.pku.edu.cn, <http://www.cs.pku.edu.cn>

Received 2004-01-16; Accepted 2004-03-29

Zhao W, Yuan CY, Liu G, Zhang SK, Wang LF. Workflow process model verification using reduction method based on P/T system. *Journal of Software*, 2004,15(10):1423~1430.

<http://www.jos.org.cn/1000-9825/15/1423.htm>

Abstract: Nowadays, workflow technology has been intensively applied in the domains of administration, production, and scientific research. For the complexity and variability of process logic, how to build a well-structured business process is a practical problem. Thus, process model analysis and improvement become one of research directions in workflow technology. Firstly, according to the modified workflow process meta-model, a P/T system based workflow model procedure net is proposed. And then, a set of simplification rules are given to verify two structural conflicts of the process model, i.e. deadlock and lack of synchronization. Finally, following these rules, a procedure net example transformed from a business process is simplified, and the result is obtained.

Key words: model verification; reduction rules; procedure net; P/T system

摘要: 目前 workflow 技术在管理、生产和科学研究等领域中已经被广泛应用。由于过程逻辑的复杂性和变化性等原因,如何建立良构的业务过程是一个比较现实的问题。因此,对过程模型进行分析和优化成为 workflow 技术的研究方向之一。首先,根据改进的工作流过程元模型,研究并提出了一种基于 P/T 系统的工作流过程模型过程网。基于过程网给出了一组化简规则,用于验证过程模型中的死锁和之同步两种结构冲突。最后,结合一个业务过程

* Supported by the National High-Tech Research and Development Plan of China under Grant No.2001AA113171 (国家高技术研究发展计划(863)); the National Grand Fundamental Research 973 Program of China under Grant No.2002CB312006 (国家重点基础研究发展规划(973)); the National Research Foundation for the Doctoral Program of Higher Education of China under Grant No.2002001016 (国家教育部博士点基金)

作者简介: 赵文(1967-),男,辽宁大连人,博士生,主要研究领域为软件工程, workflow 及相关技术;袁崇义(1941-),男,教授,博士生导师,主要研究领域为并行计算与 Petri 网, Petri 网理论及应用;刘刚(1962-),男,博士,主要研究领域为信息安全,数据挖掘;张世琨(1969-),男,博士,教授,主要研究领域为软件工程,软件体系结构;王立福(1945-),男,博士,教授,博士生导师,主要研究领域为软件工程。

实例,根据转换规则把它映射为过程网,应用规则进行化简并得到验证结果.

关键词: 模型验证;化简规则;过程网;P/T 系统

中图法分类号: TP311 文献标识码: A

workflow 是一类能够完全或者部分自动执行的业务过程,它根据一系列过程规则使得文档、信息或任务能够在不同的执行者之间传递或执行,以达到确定的业务目标. workflow 技术是实现业务过程的建模、仿真分析、优化、管理与集成,从而最终实现业务过程自动化的核心技术^[1]. 利用 workflow 技术进行建模和分析,可以规范业务过程,发现其中不合理的环节,进而对业务过程进行优化重组.

1990 年, Michael Hammer 博士首先提出了业务过程再造(business process reengineering, 简称 BPR)的概念. 20 世纪 90 年代以来,业务过程再造受到普遍关注,为政府、企事业单位管理上的变革提供了理论基础^[2]. 因此,分析业务过程、发现其中存在的问题(如过程模型中存在的瓶颈、死锁等)可以说是业务过程再造的关键.

目前,在 workflow 模型分析方面的研究取得了一定的成果. 文献[3]指出了 workflow 模型分析的不同方式,并把对 workflow 模型的分析分为 3 种类型:

- 确认(validation): 检查和测试 workflow 是否达到了预期的结果. 通常运用交互模拟技术把一些模拟用例输入系统,观察是否达到预先所期望的功能.
- 验证(verification): 检测是否正确地建立了 workflow 模型. 通常可以使用线性代数理论和基于 Petri 网的分析技术.
- 性能分析(performance analysis): 通过处理时间、服务成本、资源利用率等数据来评价业务过程满足需求的能力. 通常可以使用模拟和马尔科夫链方法.

在基于 Petri 网的模型验证方面,目前已经提出了一些相关的技术. 首先,文献[4]总结了需要验证的一些特性,例如,无死锁(deadlock freeness)、无乏同步(lack-of-synchronization freeness)、类活性(quasi-liveness)、活性(liveness)、有界性(boundedness)、起始状态(home state)等. 其中,在 workflow 过程模型中经常出现的是死锁和乏同步两种结构冲突. 主要模型验证方法包括基于图形理论的方法、基于线性代数的方法和基于状态的方法等. 其中,模型化简方法符合 Petri 网图形和严格语义特点,不受状态空间困扰,且利于机器实现. 文献[5]中提出一种基于无环有向图(DAG)的可视化验证方式和算法,并给出和使用了一系列化简规则来识别出过程模型中的结构冲突,同时也对化简过程的正确性和复杂性进行了探讨. 文献[6]提出了一种基于 Petri 网(E/N 系统)化简规则的 workflow 过程模型验证方法,并证明了所提规则的完备性和多项式时间复杂性.

根据改进的 workflow 过程定义元模型,本文研究并提出了一种基于 P/T 系统的 workflow 过程模型——过程网. 然后,基于过程网给出了一组化简规则,用于验证过程模型中的死锁和乏同步两种结构冲突. 最后,结合一个业务过程实例,根据转换规则把它映射为过程网,应用相应的化简规则进行化简并得到验证结果.

本文第 1 节讨论改进的 workflow 过程定义元模型,给出了一些核心建模元素的概念和符号,并建立了一个业务过程的实例,用于后面的模型验证. 第 2 节重点介绍了业务过程与过程网间的映射. 第 3 节给出了模型验证即化简方法的规则集,并给出了一个化简过程的实例.

1 workflow 过程模型

在面向用户的实际应用过程中,一般是通过图示、文字、符号等对业务过程模型进行描述. 首先,通过业务建模提供了一个一致的业务表现形式,即提供交流的手段,使得业务人员能更好地描述业务与表达需求,并使软件开发人员更容易理解业务,为更好地描述已有业务过程和明确需求奠定了基础. 其次,业务建模技术可以帮助业务人员和软件开发人员建立详细的业务过程模型,实现业务过程分析和设计的可视化,为规范与改进业务过程提供了基础素材. workflow 模型是对 workflow 的抽象表示,也就是对业务过程的抽象表示.

为了更好地表达 workflow 模型以适应日益复杂的业务逻辑,根据问题分离(separation of concern)原则,我们对 workflow 管理联盟制定的 workflow 过程定义元模型作了适当的改进和扩展,把汇聚、分支结构和它们的约束

(AND,OR,XOR)的说明从活动中提取出来,引入了一种新的元模型元素连接符(connector),把所有控制信息说明从活动说明中独立出来分别封装,使得变化的影响局部化.在改进的元模型中,核心元素是连接符.这样,对活动定义的修改或对活动间结构关系以及约束条件的修改变得更加容易.经过改进和扩展的元模型参见文献[7].

我们把主要的工作流过程元模型元素分为两类,即活动类和控制类.活动类主要包括活动、子过程、块等.控制类主要包括连接符、条件、转移等.这些元素的具体图形表示见表 1.

Table 1 Elements of workflow process meta-model and their graphical representation

表 1 工作流过程元模型元素及其图形表示

Begin activity	End activity	Generic activity	“AND” connector	“XOR” connector	“OR” connector	Condition	Transition

下面介绍核心建模元素的概念.

活动(activity):指业务过程中的一个逻辑步骤或任务以及相关的描述,主要可以分为:

- 开始活动(begin activity):初始化活动,业务过程的第 1 个活动,不针对具体业务环节.
- 结束活动(end activity):终结活动,表明相应业务过程的终结,不针对具体业务环节.
- 交互活动(interactive activity):一类需要实际人员参与的活动,对应前台实际的应用逻辑.
- 自动活动(automatic activity):一类由工作流引擎自动调用完成的活动,对应于后台的应用逻辑.

我们把交互活动和自动活动统称为普通活动(generic activity).

连接符(connector):活动间控制关系的描述符.用于规约像汇聚(join)、分支(split)结构这样聚合特性以及与活动相连的约束(OR,AND,XOR).

条件(condition):业务过程推进的导航依据,对应于其中的业务规则和操作的顺序.

转移(transition):活动、连接符和子过程间关系的描述.条件通常附着在转移上,以决定过程的推进.

根据以上建模元素,下面给出一个业务过程的实例(如图 1 所示),用于后面的模型验证.由于该实例是用于模型验证,所以忽略了相关活动的语义.

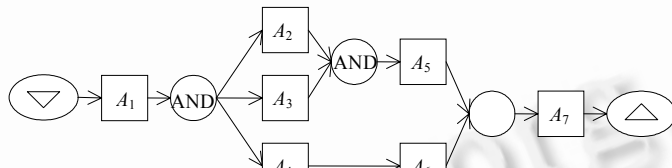


Fig.1 A business process example

图 1 一个业务过程实例

2 工作流过程模型与过程网间的映射

对工作流模型进行形式化描述,是准确定义工作流过程模型以及进行模型验证和仿真的理论基础.通过对常用工作流模式^[8]进行 Petri 网表示和同步距离^[9]描述,基于 P/T 系统提出了一种工作流模型的形式化表示方法,称为过程网(procedure net).下面简要给出过程网的定义.

定义 1. 若 $\forall t \in T : |t| \geq 1 \wedge |t^*| = 1 \wedge \forall p \in P : |p| = 1 \vee |p^*| = 1 \wedge F^+ \cap (F^-)^+ = \emptyset$, 则一个有限连结的 Petri 网 $N=(P,T,F)$ 被称为基本逻辑网.

其中,对于 $p \in P$, 若 $|p| > 0 \wedge |p^*| > 0$, 则 p 被称为一个连结符. 若 $|p| > 1 \vee |p^*| > 1$, 则连结符 p 被称作扇(fan). 若 $|p^*| > 1$, 则 p 是一个扇出(如图 2(a)所示); 若 $|p| > 1$, 则 p 是一个扇入(如图 2(b)所示).

定义 2. 若对于所有连接符 $p \in P$ 以及所有 $t \in T$:

$$(p,t) \in F \Rightarrow W(p,t) \leq |p| \wedge (t,p) \in F \Rightarrow W(t,p) \leq |p^*|,$$

则一个具有加权函数 $W:F \rightarrow \{1,2,\dots\}$ 的基本逻辑网 $N=(P,T,F)$ 构成了一个过程网 $\Sigma=(P,T,F,W)$.



Fig.2 Net structure as connector

图2 作为连接符的网结构

Σ 有两类特殊类型的库所*i*和 o_i ,库所*i*是网的唯一起始库所,即 $\bullet i = \emptyset$;过程网可以有不止一个终止库所 o_1, \dots, o_n ,即 $o_1 \bullet = \emptyset, \dots, o_n \bullet = \emptyset$.

对于连接符 $p \in P, in(p) = \sum_{t \in \bullet p} W(t, p)$ 并且 $out(p) = \sum_{t \in p \bullet} W(p, t)$.若 $in(p) = out(p)$,则 P 是一个 AND_连接符或 AND_库所,若 $in(p) \neq out(p)$,则 P 是一个 OR_连接符或 OR_库所.

过程网中的变迁对应业务过程中的活动,仅有一个输出库所,但输入库所可以多一个.输入和输出库所是连续步骤活动间的连接符.

命题1. (p 是一个连接符)

$$\begin{aligned} |\bullet p| > 1 &\Rightarrow W(t, p) = 1 \text{ for } t \in \bullet p \wedge 1 \leq W(p, \tau) \leq |\bullet p| \text{ if } \{\Sigma\} = p \bullet \\ |p \bullet| > 1 &\Rightarrow W(p, t) = 1 \text{ for } t \in p \bullet \wedge 1 \leq W(\tau, p) \leq |p \bullet| \text{ if } \{\tau\} = \bullet p \end{aligned}$$

这是定义2的一个直接结果.那么,由命题1可以明显地得出, $W(p, t)$ 或 $W(t, p)$ 是唯一可调整的连接弧上的权值(以上仅给出了与过程实例无关的过程逻辑的定义,由于篇幅所限,略去了过程语义的定义).

对业务过程模型进行分析,首先应该把它转换为过程网的表示形式.业务过程模型与过程网间的映射主要集中在活动和连结符的映射.这里,把普通活动映射为变迁,但把开始活动映射为起始库所,把结束活动映射为终止库所,把业务过程模型中的连结符映射为过程网中定义为连结符的库所.那么,连接符的基本映射规则如下:

- 如果图2中连接符的 $n = m = 1$,则表示顺序关系;
- 如果 $m = n$ 且 $m, n > 1$,则表示“与”(AND)关系;
- 如果 $m \neq n$,则表示“或”(OR)关系;
- 如果 $n = 1$,且 $m > 1$,则表示“异或”(XOR)关系.

3 化简规则及其应用

从原理上讲,基于图形化简方式的模型验证方法就是不断地运用化简规则把所有良定义的工作流过程逻辑结构从模型中化简掉,过程的最后结果是一幅特殊的图(例如,唯一库所).一个带有结构冲突的工作流过程模型不可能被完全化简.

工作流过程模型的良好规则之一是在起始节点后通过分支结构引出的所有路径,都通过在终止节点前的汇聚结构适当地被汇聚.同步结构“AND-join”被应用于汇聚“AND-split”路径,“OR-join”结构被应用于汇聚“OR-split”选择路径.针对该原则,文献[4,5]中都给出了过程模型中的两种结构冲突,即:

- 死锁(deadlock):用一个“AND-join”汇聚选择路径将导致死锁冲突.在“AND-join”结构处的死锁阻止了工作流的演进,因为“AND-join”的一个或多个前驱的转移并没有被触发.
- 乏同步(lack of synchronization):用一个“OR-join”汇聚一个“AND-split”将导致不同步.在“OR-join”处的不同步将导致后继节点被非计划地激活多次.

从解决这两种基本的结构冲突问题出发,基于过程网我们尝试提出了7条化简规则用于工作流过程模型的验证.根据化简规则的网结构,可以把它们分为3类:顺序类规则、简单分支汇聚类规则 and 多重分支类规则.下面就分别介绍各类规则.

(1) 顺序类规则:其网结构的共同特点都是表示顺序关系,区别在于,库所和变迁的位置以及是否加权.该类规则包括规则1~规则3.

规则 1. 如果一个变迁 t_1 的前集为唯一库所 p_1 , 即 $\bullet t_1 = \{p_1\}$, 其后集也为唯一库所 p_2 , 即 $t_1 \bullet = \{p_2\}$; 且库所 p_1 的前集为空, 其后集为变迁唯一变迁 t_1 , 即 $\bullet p_1 = \emptyset \wedge p_1 \bullet = \{t_1\}$; 且库所 p_2 的前集为唯一变迁 t_1 , 其后集为空, 即 $\bullet p_2 = \{t_1\} \wedge p_2 \bullet = \emptyset$; 且 $W(p_1, t_1) = W(t_1, p_2) = 1$, 那么, 可以把变迁 t_1 化简掉, p_1 和 p_2 则融合为一个孤立的新库所 p_3 , 如图 3 所示.

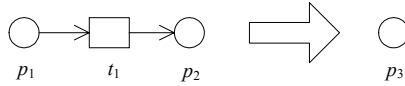


Fig.3 Reduction rule 1

图3 化简规则1

规则 2. 如果一个库所 p_1 的前集和后集为唯一变迁 t_1, t_2 , 即 $\bullet p_1 = \{t_1\} \wedge p_1 \bullet = \{t_2\}$, 且 $W(t_1, p_1) = W(p_1, t_2) = n$, t_1 的输入权值为 1, t_2 的输出权值为 m , 那么, 可以把库所 p_1 化简掉, t_1 和 t_2 则融合为一个新变迁 t_3 , t_3 的输入权值为 1, t_3 的输出权值为 m , 如图 4 所示.



Fig.4 Reduction rule 2

图4 化简规则2

规则 3. 如果一个变迁 t_1 的前集和后集为唯一库所 p_1, p_2 , 即 $\bullet t_1 = \{p_1\} \wedge t_1 \bullet = \{p_2\}$, 且 $W(p_1, t_1) = W(t_1, p_2) = n$, p_1 的输入权值为 1, p_2 的输出权值为 m , 那么, 可以把变迁 t_1 化简掉, p_1 和 p_2 则融合为一个新库所 p_3 , p_3 的输入权值为 1, 输出权值为 m , 如图 5 所示.

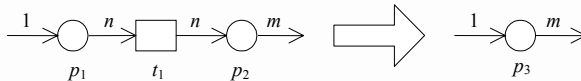


Fig.5 Reduction rule 3

图5 化简规则3

(2) 简单分支汇聚类规则: 其网结构的特点是一个简单的二分分支后接一个汇聚. 该类规则包括规则 4.

规则 4. p_1 是一个 AND_库所, $p_1 \bullet = \{t_1, t_2\} \wedge \bullet p_2 = \{t_1, t_2\}$, 库所 p_1 的输入权值为 1, 库所 p_2 的输出权值为 m , 且 $W(p_1, t_1) = n_1 \wedge W(p_1, t_2) = n_2 \wedge W(t_1, p_2) = m_1 \wedge W(t_2, p_2) = m_2$, 那么, t_1 和 t_2 则融合为一个新变迁 t_3 , 且 $W(p_1, t_3) = l_1 = n_1 + n_2 \wedge W(t_3, p_2) = l_2 = m_1 + m_2$, p_1 的输入权值仍为 1, p_2 的输出权值也仍为 m , 如图 6 所示.

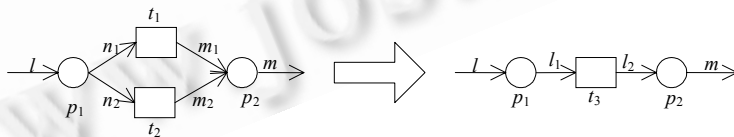


Fig.6 Reduction rule 4

图6 化简规则4

(3) 多重分支类规则: 其网结构的特点都是以一个多重分支 OR 库所开始. 包括规则 5~ 规则 7.

规则 5. p_1 是一个 OR_库所, $p_1 \bullet = \{t_1, t_2, \dots, t_k\} \wedge \bullet p_2 = \{t_1, t_2, \dots, t_k\}$, 库所 p_1 的输入权值为 l , 库所 p_2 的输出权值为 m , 若 $l = m$, 那么, 可以把变迁 t_1, t_2, \dots, t_k 化简掉, 库所 p_1, p_2 融合一个新的库所 p , 其输入输出权值分别为 1 和 m , 如图 7 所示.

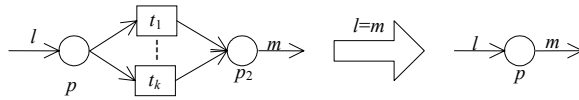


Fig.7 Reduction rule 5

图7 化简规则5

规则 6. p_1 是一个 OR_库所, $p_1 \bullet = \{t_1, t_2, \dots, t_k\} \wedge p_2 = \{t_1, t_2, \dots, t_k\}$, 库所 p_1 的输入权值为 l , 若 $l=1$, 那么, 可以把变迁 t_1, t_2, \dots, t_k 以及库所 p_2 化简掉, 使得库所 $p_1 \bullet = \emptyset$, 如图 8 所示.

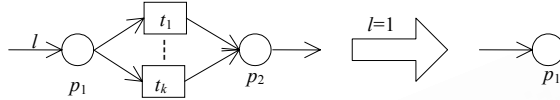


Fig.8 Reduction rule 6

图8 化简规则6

规则 7. p_0 是一个 OR_库所, $p_0 \bullet = \{t_1, t_2, \dots, t_k\}$, 且 t_1, t_2, \dots, t_k 的后集分别是唯一库所 p_1, p_2, \dots, p_k , 而 p_1, p_2, \dots, p_k 的后集均为空, 库所 p_0 的输入权值为 l , 若 $l=1$, 那么, 可以把变迁 t_1, t_2, \dots, t_k 以及库所 p_1, p_2, \dots, p_k 化简掉, 使得库所 $p_0 \bullet = \emptyset$, 如图 9 所示.

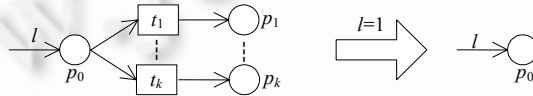


Fig.9 Reduction rule 7

图9 化简规则7

至此, 可以得到如下的结论: 如果使用图 3~图 9 所示的化简规则集, 一个过程网可以被化简为唯一库所, 则认为它不包含死锁和乏同步两种结构冲突.

下面我们就运用本文提出的 workflow 模型验证方法对一个过程模型实例进行化简. 首先, 选用本文第 1 节给出的业务过程作为模型验证的实例. 然后, 运用第 2 节的模型映射规则, 把它转换为过程网的形式, 再运用化简规则集进行化简验证.

图 10 是一个包含了结构冲突的过程网, 因为 p_1 是一个 OR_库所, 没有规则可以应用于其局部结构. 图 11 除了 $w(t_1, p_1)=3$ 以外, 与图 10 相同, 这样, p_1 是一个 AND_库所. 图 12 给出了把图 11 化简为一个独立库所的过程, 因此, 图 11 所示过程网不包含结构冲突.

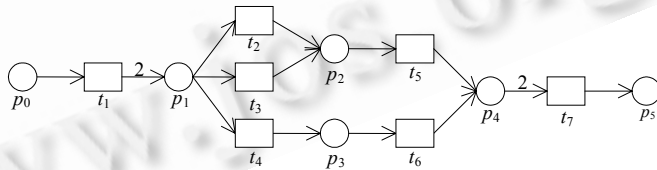


Fig.10 Not well structured

图10 包含结构冲突

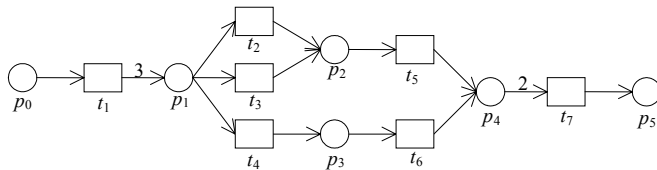


Fig.11 Well structured

图11 不包含结构冲突

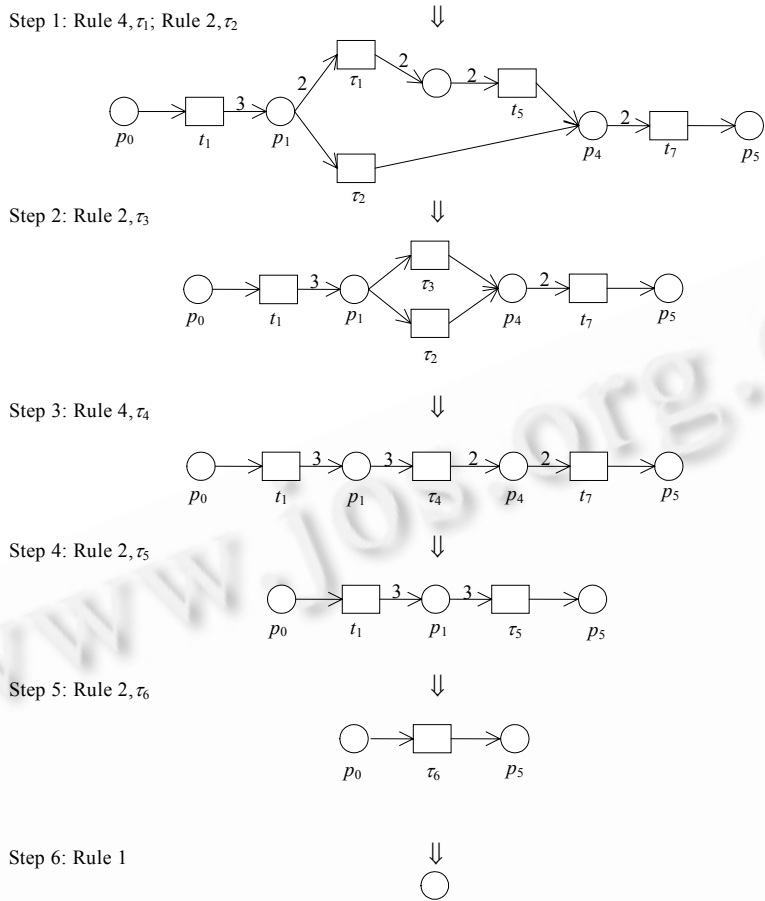


Fig. 12 A proof of Fig. 11

图 12 对图 11 的证明

4 结 语

本文采用了用于模型验证的化简方法,基于过程网提出相应的化简规则,并给出一个模型化简验证的实例.与 E/N 系统相比,P/T 系统用更少的节点元素即可以模拟同样的应用系统,相对而言也更适合于计算机自动分析工具的开发.就 workflow 模型化简验证方法来说,其效率也更高.由于本文提出的化简规则未能更多地应用于比较复杂的工作流过程模型,所以,化简规则集的完备性成为我们下一步要研究和解决的问题之一.

References:

- [1] Fan YS. Foundation of Workflow Management Technique. Beijing: Tsinghua University Press, Berlin: Springer-Verlag, 2001 (in Chinese).
- [2] Wang TM, Hu YG. BPR and Information Integration Based on Value Chain. Beijing: Tsinghua University Press, 2001 (in Chinese).
- [3] van der Aalst WMP. The application of Petri nets to workflow management. Journal of Circuits, Systems and Computers, 1998,8(1): 21~66.
- [4] Girault C, Valk R. Petri nets for System Engineering—A Guide To Modeling, Verification, and Application. Berlin: Springer-Verlag, 2003.
- [5] Orłowska ME. Analyzing process models using graph reduction techniques. Information Systems, 2000,25(2):117~134.

- [6] Li JQ, Fan Y. Workflow model verification based on Petri net reduction method. Information and Control, 2001,30(6):492~497 (in Chinese with English abstract).
- [7] Zhao W, Hu WH, Zhang SK, Wang LF. Study and application of workflow meta-model. Journal of Software, 2003,14(6): 1052~1059 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/14/1052.htm>
- [8] van der Aalst WMP, der Hofstede AHM, Kiepuszewski B, Barros AP. Workflow patterns. BETA Working Paper Series, WP 47, Eindhoven University of Technology, 2000.
- [9] Yuan CY. Principle of Petri Nets. Beijing: Publishing House of Electronics Industry, 1998 (in Chinese).

附中文参考文献:

- [1] 范玉顺. workflow管理技术基础. 北京:清华大学出版社,柏林:施普林格出版社,2001.
- [2] 王田苗,胡耀光. 基于价值链的企业过程再造与信息集成. 北京:清华大学出版社,2001.
- [6] 李建强,范玉顺. 基于Petri网化简方法的工作流模型验证. 信息与控制,2001,30(6):492~497.
- [7] 赵文,胡文惠,张世琨,王立福. 工作流元模型的研究与应用. 软件学报,2003,14(6):1052~1059. <http://www.jos.org.cn/1000-9825/14/1052.htm>
- [9] 袁崇义. Petri网原理. 北京:电子工业出版社,1998.

敬告作者

《软件学报》创刊以来,蒙国内外学术界厚爱,收到许多高质量的稿件,其中不少在发表后读者反映良好,认为本刊保持了较高的学术水平.但也有一些稿件因不符合本刊的要求而未能通过审稿.为了帮助广大作者尽快地把他们的优秀研究成果发表在我刊上,特此列举一些审稿过程中经常遇到的问题,请作者投稿时尽量予以避免,以利大作的发表.

1. 读书偶有所得,即匆忙成文,未曾注意该领域或该研究课题国内外近年来的发展情况,不引用和不比较最近文献中的同类结果,有的甚至完全不列参考文献.

2. 做了一个软件系统,详尽描述该系统的各个方面,如像工作报告,但采用的基本上是成熟技术,未与国内外同类系统比较,没有指出该系统在技术上哪几点比别人先进,为什么先进.一般来说,技术上没有创新的软件系统是没有发表价值的.

3. 提出一个新的算法,认为该算法优越,但既未从数学上证明比现有的其他算法好(例如降低复杂性),也没有用实验数据来进行对比,难以令人信服.

4. 提出一个大型软件系统的总体设想,但很粗糙,而且还没有(哪怕是部分的)实现,很难证明该设想是现实的、可行的、先进的.

5. 介绍一个现有的软件开发方法,或一个现有软件产品的结构(非作者本人开发,往往是引进的,或公司产品),甚至某一软件的使用方法.本刊不登载高级科普文章,不支持在论文中引进广告色彩.

6. 提出对软件开发或软件产业的某种观点,泛泛而论,技术含量少.本刊目前暂不开办软件论坛,只发表学术文章,但也欢迎材料丰富,反映现代软件理论或技术发展,并含有作者精辟见解的某一领域的综述文章.

7. 介绍作者做的把软件技术应用于某个领域的工作,但其中软件技术含量太少,甚至微不足道,大部分内容是其他专业领域的技术细节,这类文章宜改投其他专业刊物.

8. 其主要内容已经在其他正式学术刊物上或在正式出版物中发表过的文章,一稿多投的文章,经退稿后未作本质修改换名重投的文章.

本刊热情欢迎国内外科技界对《软件学报》踊跃投稿.为了和大家一起办好本刊,特提出以上各点敬告作者.并且欢迎广大作者和读者对本刊的各个方面,尤其是对论文的质量多多提出批评建议.