

一种面向对象 Petri 网模型的语义和行为分析

杨武^{1,2} 李晓渝² 曹泽瀚²

(重庆工学院计算机系 重庆 400050)¹ (重庆大学计算机学院 重庆 400044)²

摘要 面向对象 Petri 网根据面向对象思想的特点,对基本 Petri 网进行了扩展,引入了门、位置类型函数、多态迁移、聚集迁移以及禁止弧等概念。本文介绍了一种面向对象 Petri 网模型(Object-Oriented Petri Net, OOPN),给出了 OOPN 的基本定义和图形表示,并对 OOPN 进行了语义与行为分析。OOPN 能够成为面向对象系统的一种形式化的建模工具。

关键词 Petri, 面向对象, 建模

The Semantic and Behavioral Analysis of an Object-Oriented Petri Nets Model

YANG Wu^{1,2} LI Xiao-Yu² CAO Ze-Han²

(Department of Computer Science, Chongqing Institute of Industry and Technology, Chongqing 400050)¹

(School of Computer, Chongqing University, Chongqing 400030)²

Abstract Based on the tight combination of Petri Net theory and Object-Oriented technology, a class of high-level Petri Nets named OOPN, i. e. Object-Oriented Petri Nets, is defined, with extensions to normal Petri Nets such as gates, place-token type mapping, polymorphous transitions and collective transitions. The basic definitions and graphical denotations are introduced. The semantic and behavioral analysis of OOPN is given. OOPN can be used as formal modeling tool for Object-Oriented System.

Keywords Petri, Object-Oriented, Modeling

1 引言

面向对象技术在软件开发方面呈现出巨大的优越性并被视为解决软件危机的突破口。但是,面向对象技术在实践中,特别是在复杂系统的分析与设计中还面临许多待解决的问题。其中一个重要的问题就是需要找到能够严格地描述对象和系统动态行为特征,支持对并发性的分析与设计并能进行形式化分析的建模方法与工具。Petri 网作为一种图形化的数学工具能很好地描述并发、异步、同步、冲突等重要现象,又有严格的数学基础。用它作为系统建模和分析工具,关键在于具体应用中能妥善解决状态空间爆炸问题。

面向对象 Petri 网^[1-3]将面向对象理论和 Petri 网理论有机地结合起来,根据面向对象方法的原则对基本 Petri 网进行了扩展,在继承了 Petri 网的长处基础上,又能较有效地避免 Petri 网的状态空间爆炸问题,成为面向对象系统的建模的重要工具。

2 面向对象 Petri 网的定义

目前 OOPN 有多种不同的定义^[18],这些定义从不同的角度来描述面向对象思想与 Petri 网理论的结合,因此所定义的 OOPN 有很大差别,本文给出的仅是 OOPN 的一种定义。

面向对象 Petri 网 OOPN 是一个八元组, $OOPN = (P, T, G, F, FI, S, \psi, m_0)$, 其中:

$P = \{p_1, p_2, \dots, p_m\}$ 是位置有限集; $T = OT \cup PT \cup CT$ 是迁移(Transition)有限集, OT 、 PT 和 CT 分别为原始迁移集

(Original Transition Set)、多态迁移集(Polymorphous Transition Set)和聚集迁移集(Collective Transition Set); $G = \{GI, GO\}$ 是门,其中 GI 、 GO 分别为输入门(Input Gate)和输出门(Output Gate); F 是 $(P \times (TUG)) \cup ((TUG) \times P)$ 的有向弧集; FI 是 $P \times T$ 的禁止弧集; $S = \{S_1, S_2, \dots, S_r\}$ 为标记类型有限集; ψ 是一个映射 $\psi: P \rightarrow S$, 称为位置类型函数; m_0 是一个映射 $m_0: P \rightarrow N$, 称为初始标记状态。

OOPN 的图形表示和基本 Petri 网不同之处在于扩展部分,在 OOPN 中,用双竖线表示门,用前端带有圆圈的线段表示禁止弧,在迁移矩形顶部用英文大写字母 O 、 P 、 C 标识迁移类型(原始迁移、多态迁移、聚集迁移)。迁移条件 TC 和迁移动作 TA 标注在迁移矩形中。图 1 是 OOPN 图形表示的一个实例。

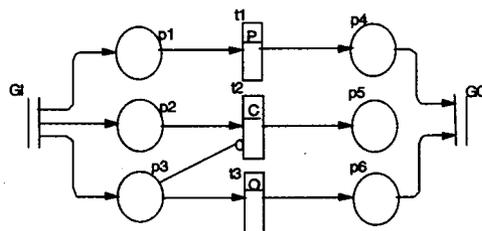


图 1 面向对象 Petri 网 OOPN 图示

3 语义分析

语义表示在实际系统建模中是必不可少的,缺乏语义表示,会令建模困难,也使得模型难以理解和维护。简单的基本

Petri 网,以及后来发展的高级网,如着色网、谓词/转移网等,都无法表示复杂的语义。在 OOPN 的定义中,OOPN 被赋予了丰富的语义表示,从而加强了 Petri 网的表达能力。下面对这些扩展的语义表示加以说明。

3.1 门

门是 OOPN 与外界环境的接口。OOPN 被输入门和输出门封装成一个单入口单出口的模块。可以认为门是一类特殊的迁移,它的发生实现 OOPN 与外界环境的信息传递。门的作用是发送和接收消息,因而门动作总和特定类型的消息有关。门动作被定义为发送或接收动作加上消息类型。如 $GI(\text{message type})$ 指 GI 接收到一个 message type 类型的消息标记, $GO(\text{message type})$ 则指 GO 发送出一个类型为 message type 的消息标记。

输入消息队列 $IMQ(\text{Input Message Queue})$ 和输出消息队列 $OMQ(\text{Output Message Queue})$ 是两个集合, $IMQ = \{y | (GI, y) \in F, y \in P\}$, $OMQ = \{y | (y, GO) \in F, y \in P\}$ 。

IMQ 和 OMQ 是位置的子集。而门完成动作,将从外界接收的消息标记按类型放入相应的 IMQ 位置(输入门),或从 OMQ 中取出队列首的消息标记发往外界(输出门)。因此, IMQ 和 OMQ 实际上是消息缓冲区。消息标记具有类型,而在 OOPN 中,一个位置只能容纳同种类型的标记。

设 OOPN 的标记状态为 m ,则门的激活和发生规则为:

(1) GI 被激活后,当且仅当 GI 接收到外界发来的消息标记,并且在 IMQ 中存在一位置能够容纳该类型的消息标记; GO 被激活,当且仅当 $\sum m(p_i) \geq 1, p_i \in OMQ$ 。

(2)门被激活立即导致门动作的发生, GI 将接受到的消息标记放入相应类型的 IMQ 位置; GO 从 OMQ 队列中取出一个特定类型的消息标记发送列外界。

由于引入了门的机制,OOPN 的内部信息被完全封装起来。从外界看,一个 OOPN 能够对特定类型的消息作出反应,完成特定的动作,发出特定的消息,而其内部的控制信息和数据信息是不可见的。门非常自然地与面向对象理论的封装和信息隐藏原则相吻合,而且提供了一种灵活、易理解的 OOPN 分层机制。

3.2 标记类型

在 OOPN 中,标记被定义为由标识及其属性组成的实体,类似一个结构化的变量。因此,标记具有了丰富的表达能力。在网的运行过程中,标记的属性可能改变,而迁移对标记属性的修改代表了特定的语义和行为。

由于标记是 OOPN 中一切数据信息和控制信息的载体,因此标记在网中的流动既是数据流,也是控制流。标记类型是对标记的抽象。一个系统的所有标记类型构成了系统的处理对象。在 OOPN 中,一般有两种标记类型,即数据标记和消息标记。在位置中标记一般按 FIFO 排队。

3.3 禁止弧

禁止弧从位置连到迁移。当禁止弧所连输入位置中有标记时,即使其所连迁移的其它输入位置都有标记并且迁移条件 TC 为真,该迁移也不会被激活。仅当禁止弧所连输入位置中没有标记时,迁移才有可能被激活。当迁移发生时,禁止弧并不从所连输入位置中移走标记,它仅起一种控制作用。禁止弧的引入使得网具有了测“零”的能力,从而获得了图灵机水平的建模能力。

禁止弧在 OOPN 中可以表达一种同步约束即优先级控制。如图 2,迁移 t_2 具有比 t_1 更高的优先级,因此,仅当 t_2 不

被激活(即 p_3 中没有标记)时, t_1 才有可能被激活。

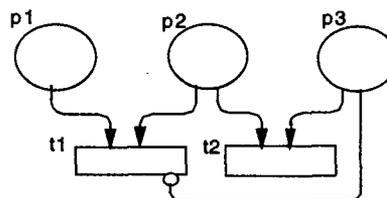


图 2 禁止弧与优先级控制

3.4 位置类型函数

由定义 1 可知,每个位置对应于某种标记类型(映射 Ψ)。一个位置中可以有多个标记,但其类型必须相同。迁移的输入位置隐含地映射于同一类型的输出位置,而一个迁移的同类型的输入位置不能多于一个,同类型的输出位置也不能多于一个,即任意 $t \in T, (p_1, p_2 \in \cdot t) \vee (p_1, p_2 \in t^*), \Psi(p_1) \neq \Psi(p_2)$ 。

在 OOPN 中,由于标记带有结构,具有一定的类型,因而能够代表可区分的物理和逻辑实体。如果仍然同基本 Petri 网一样让标记随意流动,往往使实际系统得不到正确的模拟并导致不确定性和分析复杂度的提高。引入位置类型函数的目的就是限制标记的流动,使其按照对象操作的实际输入输出语义规范起来。

3.5 迁移条件 TC 和迁移动作 TA

OOPN 是谓词/转移网,每个迁移都可以带有一个谓词,即迁移条件。只有迁移条件被满足时,迁移才有可能被激活。迁移的动作是一个过程,它规定了迁移发生时要完成的任务,包括信息处理、状态改变、事件产生等。

3.6 迁移类型

由 OOPN 的定义可知,OOPN 存在三种类型的迁移:原始迁移、多态迁移和聚集迁移。原始迁移是该 OOPN 私有的迁移,可以具有迁移动作。多态迁移和聚集迁移是实现对象继承和组装的机制,不带迁移动作。多态迁移和聚集迁移另有专文详细论述。

由于增加了上述的语义表示,OOPN 的迁移激活和发生规则与基本 Petri 网相比也有所不同。

4 OOPN 行为分析

给定一个 OOPN,设其初始标记状态为 m_0 。在 m_0 下,设有 m 个迁移被激活。任选其中的一个迁移,记为 t ,使 t 发生。 t 发生后,OOPN 的标记状态发生改变,记为 m' 。这个过程可以写为 $m(t) \rightarrow m'$ 。如果门被激活,门动作的发生同样也可能改变标记状态,写为 $m(G) \rightarrow m'$ 。由于门被激活后立即导致其动作的发生,因此门动作的发生比迁移发生具有更高的优先级。OOPN 的运行就是重复上述的过程。

由于 OOPN 是面向对象思想和 Petri 网理论的有机结合,并且在 OOPN 中引入了门的机制,因而 OOPN 具有和基本 Petri 网不同的行为特性。不失一般性,在下面的讨论中我们设 $CT = \emptyset$,并且所有迁移的迁移条件 TC 恒为真,即迁移的激活不带谓词限制。

4.1 可达性

和基本 Petri 网不同,OOPN 不再是一个孤立的网,它和外界之间存在动态的消息传递。在基本 Petri 网中,只有迁移的发生才能影响网中的标记数量和分布。而在 OOPN 中,消息的交互和迁移的发生都会影响网中的标记数量及分布。

消息的动态交互可能导致网的标记状态的无限膨胀。

用 OOPN 建立对象模型时,一个对象就是一个 OOPN,而对象和 OOPN 之间存在这样的一个状态转换 $R_i: R(m_0) \rightarrow OS$,其中 $R(m_0)$ 为 OOPN 标记状态集合, OS 为对象状态集合。利用状态转换函数 R_i 可以在 OOPN 的可达标记状态集 $R(m_0)$ 上定义一个等价关系 $R_e, m_i R_e m_j, R_i(m_i) = R_i(m_j)$ 。 $R(m_0)$ 被等价关系 R_e 划分成一些等价类 M_1, M_2, \dots, M_n 。这样,我们就将对可能无限的 OOPN 标记状态的分析,近似地转化为了对有限的标记状态等价类(即对象状态)的分析,因而 OOPN 的运行就导致了一个 OOPN 标记状态等价类序列 $M_1 \rightarrow M_2 \rightarrow \dots \rightarrow M_n$ 。如果用结点表示标记状态等价类,弧表示迁移或门动作, OOPN 不断运行可以得到一个有向图,如图 3 所示。

考察图 3,如果把迁移有限集和门动作的并集当作字母表,把包含 m_0 的标记状态等价类 $[m_0]$ 当作初态,那么 OOPN 有向图就是一个有限状态自动机。这个自动机接受的语言(迁移和门动作序列)就是 OOPN 的行为。

OOPN 向有限状态自动机的转化,实际上就是将对象的行为按照 OOPN 规定的异步和同步约束转化为顺序执行的过程。这种转化并不和并发相矛盾。

为了降低形成有向图的开销,有时可能只考虑那些关键的和感兴趣的迁移发生序列,也可以直接将对象的状态转移图近似地映射到 OOPN 有向图。

4.2 活性和死锁

OOPN 的活性和可达标记状态集紧密相关。如果 OOPN 存在死锁,即存在某个标记状态使得所有迁移都不能发生,那么显然 OOPN 不具备活性。由此,可以得到 OOPN 具备活性的一个必要条件。

如果一个 OOPN 具有活性,那么在其有向图中没有这样的结点:结点或者出度为 0,或者只有标记为门动作的自环。

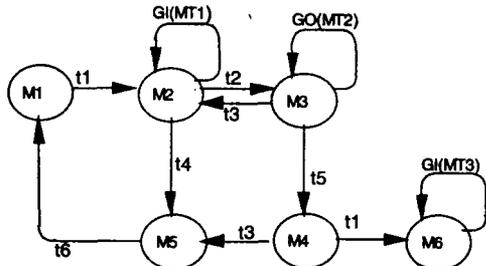


图 3 OOPN 有向图实例

4.3 有界性

外界通过输入门可以不断地向 OOPN 发送消息标记,因而输入消息队列 IMQ 中的位置所容纳的标记数量可能趋于无穷大。因此, OOPN 是一个无界(unbounded)的网。如果系统必须考虑资源(缓冲区和存贮器等)的物理容量,那么可以为 OOPN 加上这样的限制:位置和一个正整数有关,这个正整数规定位置的容量;仅当输出位置中的标记数量不会在迁移或门动作发生后超过其容量限制时,迁移或门才有可能被激活。加上这个限制后, OOPN 成为有界网,因而它表示的系统在运行中不会发生资源需求的过度膨胀。

4.4 冲突

在 OOPN 中,冲突代表了一类不确定性。有时冲突是系统的一个重要特性。例如,在某个状态下,系统可以执行 A 动作,也可以执行 B 动作,但 A 动作和 B 动作互斥,也就是

说,执行 A 动作后就不能再执行 B 动作,反之亦然。如果需要避免冲突,我们可以利用禁止弧确定迁移的优先级,也可以利用迁移条件 TC 的设置使系统在该状态下只可能有一个迁移被激活。因此, OOPN 能够灵活地表达或避免冲突。

结语 OOPN 在基本 Petri 网的基础上,引入了门、禁止弧、位置类型函数、多态迁移和聚集迁移,并将标记定义为标识及其属性组成的实体等一系列概念,大大丰富了网的语义表示和建模能力,使得 OOPN 能够有效地描述对象和系统的内部结构及行为特征。 OOPN 不仅继承了 Petri 网的长处,易表达并发、异步、同步、死锁等重要行为特征,而且具有严格的数学基础;同时,由于很好地吸取了面向对象的封装、继承和抽象等思想,有效地避免了一般 Petri 网模型难以克服的状态空间爆炸问题,大大降低了建模和分析的复杂性。因此, OOPN 能够成为面向对象系统的建模型工具。

参考文献

- 1 Lako s C A, Keen C C. Simulation with Object-Oriented Petri Nets. [Technical Report]. Department of Computer Science, University of Tasmania, Australia, April, 1994
- 2 Camurri A, Franchi P, Vitale M. Object-Oriented Approach to High-Level Petri Nets. Microprocessing and Microprogramming. 1992, 35
- 3 Keen C D, Lako s C A. A Methodology for the Construction of Simulation Models Using Object-Oriented Petri Nets. [Technical Report]. Depart of Computer Science, University of Tasmania, Australia, April 1994
- 4 Hafedh M. On Behavioral Descriptions in Object-Oriented Modeling. J. Systems Software, 1996, 34
- 5 Jackson R B, Embley D W, Woodfield S N. Developing Formal Object-Oriented Requirements Specifications; a Model, tool and Technique. Information Systems, 1995, 20(4)
- 6 Tsai J J P, Yang S J, Chang Y-H. Timing Constraint Petri Nets and Their Application to Schedulability Analysis of Real-Time System Specifications. IEEE Trans. On Software Engineering, 1995, 21(1)
- 7 Billington J, Wheeler G R, Wilbur-ham M C. Protean; a High-level Petri Net Tool for the Specification and Verification of Communication Protocols. IEEE Trans. On Software Engineering, 1998, 14(3)
- 8 Kemper P. Numerical Analysis of Superposed GSPNs. IEEE Trans. On Software Engineering, 1996, 22(9)
- 9 Rossano. Efficient Discrete-Event Simulation of Colored Petri Nets. IEEE Trans. On Software Engineering, 1996, 22(9)
- 10 Sereno M. Approximate Mean Value Analysis for Stochastic Marked Graphs. IEEE Trans. On Software Engineering, 1996, 22(9)
- 11 Yen HSU-Chun. On the Regularity of Petri Net Languages. Information and Computation, 1996(12)
- 12 Kung D C. An Executable Visul Formalism for Object-Oriented Conceptual Modeling. J. Systems Software, 1995(31)
- 13 Hull M E C, O'Donoghue P G. Timed Petri Net Approach to Performance Modeling with the MOON Method. Software Engineering Journal, 1994
- 14 Saldhana J A, Shatz S M, Hu Z. Formalization of object behavior and interactions from UML models. Intl. Journal of Software Engineering and KnowledgeEngineering, 2001, 11(6)
- 15 蒋昌俊. 非确定并发系统设计的 Petri 网形式化方法. 系统仿真学报, 1995(9)
- 16 桂志波, 郑应平. 一类扩展 Petri 网的建模仿真分析. 系统仿真学报, 1996(9)
- 17 周莹新, 艾波. 软件体系结构建模研究. 软件学报, 1998, 9(11)
- 18 范玉顺, 张军. 面向对象的 Petri 网方法及其在软件工程中的应用. 计算机应用, 1998, 18(9)
- 19 骆华俊, 唐稚松, 郑建丹. 可视化体系结构描述语言 XYZ/ADL. 软件学报, 2000, 11(8)