并发系统中谓词行为图的行为时序逻辑表达*

黄贻望[1,2†], 袁 科[1,3]

(1. 铜仁学院 数学与计算机科学系, 贵州 铜仁 550025; 2. 武汉大学 计算机学院 软件工程国家重点实验室, 武汉 430072; 3. 南开大学 信息技术科学学院, 天津 300071)

摘 要: 行为时序逻辑(TLA)组合时序逻辑与行为逻辑,可以对并发系统进行描述与验证,它引入动作和行为的概念,使得系统和属性可用它的规约公式表示,但存在用 TLA 描述复杂系统时 TLA 公式复杂且难以理解的不足。类似于状态转移图,对于并发转移可以用谓词行为图进行图形化表示,谓词行为图与行为时序逻辑规约具有相同的表达能力。介绍行为时序逻辑的语法、语义及简单推理规则,用一个简单的实例说明使用谓词行为图去描述并发转移系统的有效性,并用系统规约的 TLA 公式对谓词行为图表达能力进行证明,表明两者具有等价性,为描述和分析并发转换系统提供了一种可行的方法。

关键词: 并发性; 规约; 谓词行为图; 行为时序逻辑

中图分类号: TP311 **文献标志码**: A **文章编号**: 1001-3695(2013)09-2752-03

doi:10.3969/j.issn.1001-3695.2013.09.048

Temporal logic of actions' expression of predicate behavior graph in concurrent system

HUANG Yi-wang[1,2†], YUAN Ke[1,3]

(1. *Dept. of Mathematics & Computer Science, Tongren College, Tongren Guizhou* 550025, *China*; 2. *State Key Laboratory of Software Engineering, College of Computer, Wuhan University, Wuhan* 430072, *China*; 3. *College of Information Technology, Nankai University, Tianjin* 300071, *China*)

Abstract: Temporal logic of actions is a logic which combines the temporal logic and the logic of actions, describing and validating the concurrent systems with this logic. TLA can express the system and the corresponding properties by adding actions and behaviors, it's harder to understanding the complexity systems' specification with the TLA formula. As same as the state transition graph, predicate behavior graph will be used to describe the concurrent systems. First, this paper introduced the grammar and semantics of temporal logic of actions and how to description the specification of concurrent systems with predicate behavior graphs, then proven the correctness of this describing method.

Key words: concurrent; reduction; predicate behavior graph; temporal logic of actions(TLA)

0 引言

为了形式化表示并发转移系统,可以用标记转移系统来表示,然后讨论这个系统的正确性。但对于复杂的系统来说,它的逻辑规范复杂且很难获得。针对上述问题,Lamport 等人[1,2]提出了一种逻辑,即行为时序逻辑 TLA(temporal logic of actions),它能在一种语言中同时表达模型程序与系统属性。在 TLA 中,一个规约就是描述系统所有可能正确行为的逻辑公式。

图形描述具有直观和易读的特点,它有助于理解一个系统某个特定的简单模块规格。因此,使用谓词行为图描述并发转移系统,这种图类似于状态转移图,通过 TLA 规约公式证明这种谓词行为图的有效性,即一个表示公式 D 的谓词行为图[3]是一个规约为 S 系统的正确描述当且仅当 $S \Rightarrow D$,从而可以得出这个图是该系统规约的正确表示。

1 TLA 的语法和语义

TLA 将一个系统描述成一个 TLA 公式(formula),该公式为真当且仅当该系统所有可能的执行(每一个执行对应一个行为)都能满足该公式为真。

其基本语法定义[1,2,4,5]如下:

$$\text{formula} \triangleq \langle \text{predicate} \rangle \mid \Box[\langle \text{action} \rangle]_{\langle \text{state function} \rangle} \mid \neg \langle \text{formula} \rangle \wedge \langle \text{formula L} \rangle$$

收稿日期: 2012-10-23; **修回日期**: 2012-12-08 **基金项目**: 中央高校基本科研业务费专项资金资助项目(2012211020201); 贵州省科学技术厅、铜仁市科学技术局、铜仁学院联合基金资助项目(黔科合 J 字 LKT[2012]04 号, 黔科合 J 字 LKT[2012]24 号); 铜仁学院科研启动基金资助项目(TS10013)

作者简介: 黄贻望(1978-),男(通信作者),湖南怀化人,博士研究生,主要研究方向为服务计算、形式化方法(hywcxq-1@163.com); 袁科(1982-),男,河南南阳人,博士研究生,主要研究方向为信息安全、光学密码.

$|\Box\langle formula\rangle$

其中:〈action〉定义为布尔表述式,由常量符号、变量及原本变量组成;〈predicate〉定义为不包含原本变量的〈action〉|Enabled〈action〉;〈state function〉定义为非布尔表达式,由常量符号及变量组成。

TLA 的标准规约形式是 $Init\wedge\Box[N]_v\wedge L$。其中:Init 是一个谓词;$N$ 是一个行动;v 是一个状态函数;L 是一个公平性的合取式。这个公式是某个行为的断言:初始状态 Init 为真;行为中每一步都是 N 或保持 v 不变的步,保持公平性 L。公平性条件则是保证有些行动一定得发生。

TLA 的语义解释是基于行为(behavior)、状态(state)和动作(action) 这三个概念的。行为表示一个无限的状态序列;状态是 TLA 中变量到具体变量值的一个映射,其语义含义是将内容赋值给逻辑中的每个语法对象;动作代表了当前状态与下一状态间的关系。

基本 TLA 语义定义为

$$S[[f]]\triangleq f(\forall v':s[[v]]/v)$$

$$S[[A]]t\triangleq A(\forall v':s[[v]]/v,t[[v]]/v')$$

$$|=A\triangleq\forall s,t\in St:s[[A]]t$$

$$S[[\text{Enabled}]]\triangleq\exists t\in St:s[[A]]t$$

2　谓词行为图

2.1　非形式化描述

以一个具有 n 个布尔变量的输入 in[1],…,in[n]和输出 out[1],…,out[n]的函数为例,如图 1 所示。

由图 1 可知,可以把组成这个实例的环境都看做是一个封闭系统。当系统处于初始状态时,所有的输入和输出都是相等的,当所有的输入均为 0 时,则输出为 0;当所有的输入为 1 时,则输出为 1。在某个输入发生改变后,其值要保留到输出发生改变。

图 2 为布尔变量输入的值及其环境的谓词行为图。

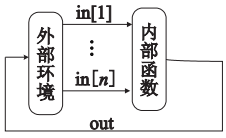

图1　布尔变量函数转换系统

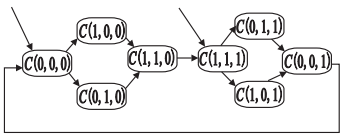

图2　谓词行为图

其相应的 TLA 公式如下:

$$\begin{aligned}&formula\triangleq\\&\wedge C(0,0,0)\vee C(1,1,1)\\&\wedge\Box[C(0,0,0)\Rightarrow C(1,0,0)'\vee C(0,1,0)']_{\langle in[1],in[2],out\rangle}\\&\wedge\Box[C(1,0,0)\Rightarrow C(1,1,0)']_{\langle in[1],in[2],out\rangle}\\&\cdots\\&\wedge\Box[C(0,0,1)\Rightarrow C(0,0,0)']_{\langle in[1],in[2],out\rangle}\end{aligned}$$

这里:

$$C(i,j,k)\triangleq(in[1]=i)\wedge(in[2]=j)\wedge(out=k)$$

图 2 中的 $C(0,0,0)$、$C(1,1,1)$ 两个节点表示初始节点,表示系统从 $C(0,0,0)$ 或 $C(1,1,1)$ 两个节点开始,两个节点之间的箭头表示可能会发生的状态转移。它们是由三元组〈in[1],in[2],out〉中值的改变而发生转移的,同时由于环境变量的改变而引起的状态转移,这时这个三元组中的元素都未发生改变。

从图 2 可以看出,它看起来很像是状态转移图,然而这里不用传统的有穷状态机[6~8]来解释谓词行为图,而是用 TLA 公式来解释的。这些公式都有形如:Init$\wedge\wedge_o F_o$,Init 指的是系统一个初始状态谓词和$\wedge_o F_o$ 每个节点 o 的公式 F_o 的合取式,Init = $C(0,0,0)\vee C(1,1,1)$。每个 F_o 描述的是从节点 o 开始所有可能的状态改变。

一个谓词行为图代表的是一个安全属性,不包括任何公平性条件。则 n 元布尔变量输入的 TLA 公式规约如下:

$$\begin{aligned}&Init_C\triangleq\wedge out\in\{0,1\}\\&\wedge in=[i\in\{1,\cdots,n\}\mapsto out]\\&input(i)\triangleq\wedge in[i]=out\\&\wedge in'=in\ \text{except}\ ![i]=1-in[i]]\\&\wedge out'=out\\&output\triangleq\wedge\forall i\in\{1,\cdots,n\}:in[i]\neq out\\&\wedge out'=1-out\\&\wedge in'=in\\&next\triangleq output\vee\exists i\in\{1,\cdots,n\}:input(i)\\&\Pi_C\triangleq Init_C\Box[next]_{\langle in,out\rangle}WF_{\langle in,out\rangle}(output)\end{aligned}$$

通过定义域{1,…,n}变量为 in 的值的函数来表示形式化这个输入序列,则 TLA 规约解释如下:

a) $init_C$。它是一个状态谓词断言的输出,要么是0,要么是1,并且输入是后个定义域为{1,…,n}的函数,即对于$\forall i\in\{1,\cdots,n\}$,有 in[$i$] = out。

b) input(i)。它是一个行动,在 in[i] = out 时使能。为强化 in[i],则当$j\neq i$时,使 in[j]和 out 不改变。

c) output。它是一个行动,使能当且仅当对于所有 in[i] ≠ out,为强化 out 使 in 不改变。

d) next。它是一个行动,是所有 output 和 input(i)($\forall i\in\{1,\cdots,n\}$)行动的析取式,所以 next 是一个 output 步或者是 input(i)步。

e) Π_C。它为规约的时序式,断言:(a) 系统的初始状态是 $Init_C$;(b)每步都是 next 或者哑步;(c)在 output 步没有出现之前,output 不能永远使能。这个公平性条件保证输出发生改变(因为谓词行为图描述的仅是安全属性,公平性条件与谓词行为图的表示无关)。

这个规约 Π_C 简短且精确,然而它不像谓词行为图那样容易理解,因而有必要用谓词行为图去解释这个规约,如图 3 所示。

图 3 是状态函数为〈in[i],out〉的谓词行为图,它意味着随着状态函数〈in[i],out〉的变化而发生的转移,也是规约 Π_C 的行为图。该行为图表明转换函数中的第 i 个输入与其输出之间的同步。

从图 3 的说明可以得出这个规约等价的谓词行为图,如图 4 所示。

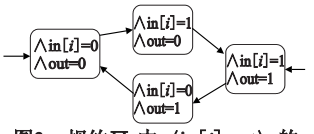

图3　规约Π_C中〈in[i],out〉的谓词行为图

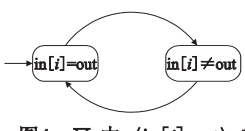

图4　Π_C中〈in[i],out〉的等价谓词行为图

因此,对同一个规约可以用多个不同谓词行为图去描述。在实际规约中,不采用图对系统进行规约,但可用直观图有助

于解释规约,希望系统包括的细节越少越好,通过画出多种不同的谓词行为图来解释系统的规约。

2.2 形式化描述

首先精确定义谓词行为图的 TLA 公式。形式上,一个谓词行为图包含一个有向图、一个从初始节点开始的节点子集。这里每一个节点都以谓词来标记,而每个边都以动作标记。假定一个给定状态函数 v 的谓词行为图并作以下解释:N 为节点集;I 为初始节点集;$E(n)$ 为从节点 n 出发的边集;$d(e)$ 为边 e 的目的节点;P_n 为标记节点 n 的谓词;ε_e 为标记边 e 的行动。

通过以下的定义得到公式△:

$$\mathrm{Init}_{\triangle} \triangleq \exists n \in I: P_n$$

$$A_n \triangleq \exists e \in E(n): \varepsilon_e \wedge P'_{d(e)}$$

$$\triangle \triangleq \mathrm{Init}_{\triangle} \wedge \forall n \in N: \Box[P_n \Rightarrow A_n]_v$$

当没有确定的标签附属边 e,则认为 ε_e 为真。当没有明确指定初始节点集,本文把 I 当做是 N,依据常用空集的习惯,A_n 当且仅当没有从节点 n 出发的边。

2.3 谓词行为图的证明

通常所说规约 Π 的谓词行为图表明 Π 蕴涵代表谓词行为图的 TLA 公式△,公式 Π 一般具有 $\mathrm{Init}_{\Pi} \wedge \Box[M]_u \wedge L$ 的形式,这里的 L 是公平性条件,公式 $\triangle \equiv \mathrm{Init}_{\triangle} \wedge \forall n \in N: \Box[P_n \Rightarrow A_n]_v$。为证明 $\Pi \Rightarrow \triangle$,需证明:

a) $\mathrm{Init}_{\Pi} \Rightarrow \mathrm{Init}_{\triangle}$;

b) 对每个节点 n,$\mathrm{Init}_{\Pi} \wedge \Box[P_n \Rightarrow A_n]_v$。

条件 a) 是对谓词的断言,这个很容易证明。为证明条件 b),通常要找到一个不变量 Inv,使得 $\mathrm{Init}_{\Pi} \wedge \Box[M]_u \Rightarrow \Box \mathrm{Inv}$,因此 $\Pi \Rightarrow \Box[M \wedge \mathrm{Inv}]_u$,通过 $[M \wedge \mathrm{Inv}]_u \Rightarrow [P_n \Rightarrow A_n]_v$ 对每个节点 n,通常 u 和 v 是元组,且每个 v 的组件是 u 的组件。在这种情况下,则需以下公式成立:对每个节点 n,定义动作 A_n,有 $M \wedge \mathrm{Inv} \Rightarrow [P_n \Rightarrow A_n]_v$,从而证明对每个节点 n:

$$P_n \wedge M \wedge Inv \Rightarrow (\exists m \in E(n): \varepsilon_m \wedge P'_{d(m)}) \vee (v' = v)$$

断言 P_n 且 Inv 为真是一个 M 步,同时改变 v 是一个 ε_m,最后对从节点 n 出发的某些边,以满足 $P_{d(m)}$ 的状态结束。

3 结束语

本文主要针对使用 TLA 公式规约复杂转移系统时存在复杂难以理解的不足,提出使用谓词行为图对转移系统进行描述,并用 TLA 规约进行解释与证明,说明了 TLA 对系统的规约与使用谓词行为图描述系统具有等价性。尽管谓词行为图具有直观易懂的优点,但在处理复杂并发转移系统时存在图形复杂。因此,需要将复杂系统分解成简单模块,使用谓词行为图分别对模块进行描述,合并讨论整个系统的正确性,这是下一步的研究工作。

参考文献:

[1] LAMPORT L. The temporal logic of actions[J]. ACM Transactions Programming Languages and Systems,1994,16(5):872-923.

[2] LAMPORT L, MALKHI D, ZHOU Li-dong. Reconfiguring a state machine[J]. SIGACT News,2010,41(1):63-73.

[3] LAMPORT L. Specifying systems [M]. Boston: Addison-Wesley, 2003.

[4] 唐郑熠,李均涛,李祥. 行为时序逻辑中公平性的研究与完善[J]. 计算机应用研究,2010,27(5):1788-1790.

[5] 夏薇,姚益平,慕晓冬. 基于 TLA 的事件图模型形式化验证方法[J]. 计算机应用研究,2011,28(11):4171-4173,4187.

[6] KAYNAR D,LYNCH N,SEGALA R, *et al*. The theory of timed I/O automata[M]. [S. l.]:Morgan & Claypool Publishers,2005.

[7] CHENG Hong, LO D, ZHOU Yang, *et al*. Identifying bug signatures using discriminative graph mining[C]//Proc of the 18th International Symposium on Software Testing and Analysis. 2009:141-152.

[8] MARCZAK W R, ALVARO P, CONWAY N, *et al*. Confluence analysis for distributed programs: a model-theoretic approach technical report, No UCB/EECS-2011-154 [R]. UC Berkeley: EECS, 2011.

[9] GUREVICH Y. Sequential abstract state machines capture sequential algorithms[J]. ACM Trans on Computational Logic,2000,1(1):77-111.

[10] 黄贻望,万良,李祥. 基于 TLA 的 NS 安全协议分析及检测[J]. 计算机工程与科学,2010,32(7):38-41.

(上接第 2751 页)

[6] 梁斌,陈敏,缪柏其,等. 基于 LARS-LASSO 的指数跟踪及其在股指期货套利策略中的应用[J]. 数理统计与管理,2011,30(6):1104-1114.

[7] HALSTEAD M H. Elements of software science[M]. New York: Elsevier North Holland,1977.

[8] McCABE T J. A complexity measure[J]. IEEE Trans on Software Engineering,1976,SE-2(4):308-320.

[9] 李轩,郝克刚,葛玮. 面向对象软件度量的分析和研究[J]. 计算机技术与发展,2006,16(11):38-41.

[10] HASTIE T, TAYLOR J, TIBSHIRANI R, *et al*. Forward stagewise regression and the monotone lasso[J]. Electronic Journal of Statistic,2007,16(1):1-29.

[11] 何俊学. 基于支持向量机的软件可靠性模型研究[D]. 兰州:兰州理工大学,2009.

[12] EFRON B,HASTIE T,JOHNSTONE I,*et al*. Least angle regression [J]. Journal of the Institute of Mathematical Statistics,2004,32(2):407-499.

[13] 姜慧研,宗茂,刘相莹. 基于 AC0-SVM 的软件缺陷预测模型的研究[J]. 计算机学报,2011,34(6):1148-1154.

[14] GRAY D, BOWES D, DAVEY N,*et al*. The misuse of the NASA metrics data program data sets for automated software defect prediction [C]//Proc of the 15th Annual Conference on Evaluation & Assessment in Software Engineering. 2011:96-103.

[15] WANG Xu-hui,SHU Ping,CAO Li,*et al*. A ROC curve method for performance evaluation of support vector machine with optimization strategy [C]//Proc of International Forum on Computer Science. Technology and Applications. 2009:117-120.

[16] 姚全珠,宋志理,彭程. 基于 LDA 模型的文本分类研究[J]. 计算机工程与应用,2011,47(13):150-154.

[17] 衷湾,余都,江顺亮. 基于 CA 的离散粒子系统仿真和显示研究[J]. 计算机与现代化,2011(1):1-5,10.

[18] 林盾,陈俐. BP 神经网络在模拟非线性系统输出中的应用[J]. 武汉理工大学学报,2003,27(5):731-734.

[19] BRIAND L C,MELO W L,WUST J. Accessing the applicability of fault-proneness models across object-oriented software projects[J]. IEEE Trans on Software Engineering,2002,28(7):706-720.