软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2012,23(11):2835−2861 [doi: 10.3724/SP.J.1001.2012.04304]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

多智体系统中约简状态空间的限界模型检测算法*

周从华+, 叶 萌, 王昌达, 刘志锋

(江苏大学 计算机科学与通信工程学院,江苏 镇江 212013)

Bounded Model Checking Algorithm to Reduce the State Space in Multi-Agent Systems

ZHOU Cong-Hua+, YE Meng, WANG Chang-Da, LIU Zhi-Feng

(School of Computer Science and Telecommunication Engineering, Jiangsu University, Zhenjiang 212013, China)

+ Corresponding author: E-mail: chzhou@ujs.edu.cn

Zhou CH, Ye M, Wang CD, Liu ZF. Bounded model checking algorithm to reduce the state space in multi-agent systems. *Journal of Software*, 2012,23(11):2835–2861 (in Chinese). http://www.jos.org.cn/1000-9825/4304.htm

Abstract: In order to specify properties relating to probability, real time, and knowledge on multi-agent systems, a logic system PTCTLK is proposed. Model checking is an automatic technique for checking if a multi-agent system satisfies a PTCTLK formula. The state explosion problem is the key obstacle in checking the feasibility of the model. In this paper, a bounded model checking algorithm for PTCTLK is proposed. First the model checking of PTCTLK is reduced to the model checking of PBTLK, which does not contain real time operators. Second, the bounded semantics of PBTLK is defined and its correctness is proven. Third, the bounded model checking procedure of PBTLK is transformed into a linear equation. Finally, the paper discusses the law of increasing of probability measure, and some termination criterions are given. The case study shows that compared with the unbounded model checking, if the length of the witness is short, then the bounded model checking needs less space.

Key words: multi-agent system; model checking; bounded model checking; state space explosion

摘 要: 为了形式化描述多智体系统中与概率、实时、知识相关的性质,提出了一种概率实时认知逻辑 PTCTLK.模型检测是验证多智体系统是否满足 PTCTLK 公式的主要技术,状态空间爆炸是该技术实用化的主要瓶颈,为此提出一种PTCTLK的限界模型检测算法.其基本思想是,在有限的局部可达空间中逐步搜索属性成立的证据,从而达到约简状态空间的目的.首先,将 PTCTLK 的模型检测问题转换为无实时算子的 PBTLK 的模型检测问题;其次,定义 PBTLK 的限界语义,并证明其正确性;然后,设计基于线性方程组求解的限界模型检测算法;最后,依据概率度量的演化规律,探索检测过程终止的判别准则.实例研究结果表明,与无界模型检测相比,在属性为真的证据较短的情况下,限界模型检测完成验证所需空间更小.

关键词: 多智体系统;模型检测;限界模型检测;状态空间爆炸

中图法分类号: TP18 **文献标识码**: A

* 基金项目: 国家自然科学基金(61003288, 61111130184); 教育部博士点基金(20093227110005); 江苏省自然科学基金(BK2010192)

收稿时间: 2012-06-04; 修改时间: 2012-08-12; 定稿时间: 2012-08-21

模型检测自 20 世纪 80 年代被提出以来,已经成为最成功的自动化验证技术之一.目前,模型检测被广泛应用于硬件、通信协议、安全协议、分布式算法的正确性证明与可靠性分析,例如,McMillan 利用模型检测技术成功地验证了微处理器的乱序执行单元[1]和多处理机的 Cache 一致性协议[2].Kwiatkowska 等人验证了 CSMA/CD 协议和 FireWire root contention 协议[3].在模型检测中,系统一般被模型化为一个有穷状态转换系统,如 Kripke 结构、有界 Petri 网等,检测的属性一般利用时态逻辑公式描述,典型的有计算树时态逻辑 CTL (computation tree logic)[4]、线性时态逻辑 LTL(linear temporal logic)[5].模型检测主要通过遍历有穷状态转换系统的全局空间来验证属性是否成立.

知识推理[6]一直是多个领域的研究人员积极探索的课题,比如哲学、经济、人工智能、分布式系统等.特别是在分布式系统领域,时态认知逻辑能够更准确地描述系统和协议的规范.因此,自 1991 年 Halpern 和 Vardi 提出利用模型检测技术完成时态认知逻辑的演绎推理之后[7],模型检测时态认知逻辑一直是一个重要的研究领域,并且在多智体系统中得到了广泛的应用.例如,在文献[8]中,Meyden 等人将每一个保密家视为一个智体,就餐的多位保密家构成了一个多智体系统,使用时态认知逻辑刻画了协议的匿名性,利用模型检测技术判断出逻辑公式成立,从而验证了保密家协议的匿名性;在文献[9]中,骆翔宇等人将铁路控制系统视为一个多智体系统,利用模型检测技术验证了控制系统的活性.

模型检测时态认知逻辑的研究主要围绕 3 个方面进行,包括模型检测算法、系统和属性规范、状态空间约简技术.给定一个系统模型和一个逻辑公式描述的属性规约,检测算法主要判断系统是否满足规约.例如,Hoek 等人提出了一种将时态认知逻辑 CKLn 的模型检测问题转化为 LTL 模型检测问题的方法[10].规范主要研究不同特性的逻辑刻画,基本的认知时态逻辑包括 CKLn[11],CTLK(computation tree logic of knowledge)[12].然而在实际应用中,有时必须考虑实时性,比如在火车穿越控制系统中,火车进站的信号发出后,安全门必须在 50 秒内关闭等等.为此,Lomuscio 等学者提出了实时认知逻辑 TECTLK[13].在另外一些重要领域,对某些事件发生的可能性进行推理也比较重要,比如,“事件 $E1$ 发生的概率小于 1/3”、“在请求发出之后两秒内得到响应的概率不低于 98%”等等.这些规范可以通过在时态认知逻辑中引入表示概率分布的算子得到表示,为此,Ferreira 等学者提出了概率认知逻辑 P_FKD45[14],Wan W 等人提出了概率认知逻辑 PCTLK(probabilistic computation tree logic of knowledge)[15].本文的第 1 项工作是提出一种概率实时认知逻辑 PTCTLK(probabilistic time computation tree logic of knowledge),从而可以对概率、实时以及知识进行推理.这种集成是自然的和平凡的,本文的主要创新点在于提出能够同时处理概率度量和实时算子的限界检测技术.

模型检测通过遍历全局空间完成系统的验证.然而对于实际的设计,系统中状态的数目随着并发分量的增加呈指数级增长,这就是模型检测中最著名的状态空间爆炸问题.认知逻辑的模型检测面临同样的问题,为此,多位学者研究了状态空间约简技术,主要包括符号化计算[8]、抽象[16,17]、偏序归约[18]、限界检测[12,19]等等.其中,限界模型检测的基本思想是:在有限的局部空间中逐步搜索属性成立的证据或者失效的反例,从而达到约简状态空间的目的.

一般来讲,限界模型检测有 3 个核心问题,即限界语义的定义、限界模型检测算法、判断检测过程终止的准则.概率实时认知逻辑 PTCTLK 中的概率度量算子给限界检测带来了很多新的理论问题,本文致力于解决这些问题,并对新问题展开系统的研究,具体工作包括 4 个方面:

1) 将 PTCTLK 的模型检测问题转换为无实时算子的 PBTLK(probabilistic branching temporal logic of knowledge)的模型检测问题.
2) 定义了 PBTLK 的限界语义,并证明了其正确性.
3) 设计了基于线性方程组求解的限界模型检测算法,即将 PBTLK 的限界模型检测问题转换为线性方程组的求解问题.
4) 说明以路径长度作为检测过程终止的判别条件已经失效.我们基于数值计算中牛顿迭代法使用的迭代过程终止判断准则,设计了一系列的终止性判别准则,并分析了各种准则适用的场景.

另外,针对线性方程组的特点,给出了变元求解的次序,从而避免不必要的迭代运算.实例研究结果表明,与

传统的限界模型检测一样,PTCTLK 的限界模型检测是一种前向搜索状态空间的方法,在属性为真的证据较短的情况下,完成验证所需内存空间较少.

1 相关工作

1999 年,Biere A 首次提出将 LTL 的模型检测问题转换为命题公式的可满足性判定问题[20],这被认为是限界模型检测的起源.限界模型检测的基本思想是只遍历足以用来验证某一规范的部分状态空间.其优点是不会遇到状态爆炸问题,并且能够快速获取最小长度的反例,内存消耗远小于基于有序二叉决策图 OBDD(ordered binary decision diagram)的方法,而且无须对变量进行静态或动态排序.

2002 年,Penczek 将限界检测技术应用于 CTL 的全称片断的验证[21].2003 年,Penczek 等人进一步提出了多智体系统中验证时态认知逻辑 ACTLK(CTLK 的全称片断)的限界模型检测方法[12],并开发了相应的限界检测工具 BMCIS.Luo 等人在时态逻辑 CTL*的语言中引入认知模态词,得到一个新的时态认知逻辑 ECKLn,并展示了相应的限界模型检测算法[19].Lomuscio 等人将实时性引入到认知逻辑中,得到了一种实时认知逻辑 TECTLK,并提出了 TECTLK 的限界模型检测算法[13].

上述不同逻辑系统的限界检测,本质上是为验证的属性寻找反例的过程,并将反例的存在性转换为命题公式的可满足性判定问题.造成能够转换为命题公式可满足性判定问题的关键在于这些反例仅仅涉及状态转换,转换关系可以用命题公式编码.而概率算子的反例由于在路径的转换过程中涉及概率度量,利用命题公式很难进行编码.因此,概率算子的限界检测有很多新的特性,这些新特性使得我们必须对概率算子的限界检测进行系统的研究.

举例如下:公式 $F\phi$的证据是一条有穷路径 $s_0,s_1,\ldots,s_k$,其中,s_k 满足ϕ,因此只需对 s_i,s_{i+1} 之间的转换关系以及 s_k 满足ϕ进行编码.而对于公式 $P_{\geqslant\frac{1}{3}}(F\phi)$,证据是路径的集合,比如 $\{s_0\xrightarrow{1/4}s_1\xrightarrow{1}s_2,s_0\xrightarrow{1/5}s_1\xrightarrow{1}s_3\}$,其中,$s_2$,$s_3$ 满足ϕ.因为事先无法预知状态之间的转换概率,所以无法计算集合当中包含多少条路径,从而无法使用命题公式编码.

此外,传统的限界检测中会对路径的长度设置一个界,使得如果存在反例则必存在长度不超过界的反例,从而保证了限界检测技术的完备性.但是对于概率算子,这种方法将失效.考察如图 1 所示的例子,验证的属性是初始状态 s_0 是否满足 $P_{\geqslant 1}(F\phi)$.显然,s_0 满足 $P_{\geqslant 1}(F\phi)$,但是随着步长的增加(从步长 0 开始),我们得到这样一个概率度量序列 $0,\frac{2}{3},\frac{2}{3},\frac{8}{9},\frac{8}{9},\frac{26}{27},\frac{26}{27},\ldots,1-\left(\frac{1}{3}\right)^{\left\lceil\frac{k-1}{2}\right\rceil},\ldots$ 因此,在事先不能确定是否已经遍历全局空间的情况下,无论步长如何增长,始终不能得到真实的概率度量 1.此实例说明,以设置路径长度的上限作为判断算法终止的标准已经失效,必须设计新的标准等等.总之,概率度量算子为限界检测技术带来了很多新问题.本文将对这些新问题进行详细的研究.

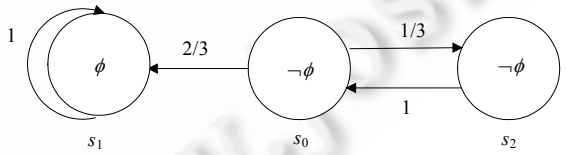

Fig.1 A simple probabilistic transition system

图 1 一个简单的概率转换系统

2 概率实时解释系统

2.1 概率分布与度量

定义 1(概率分布)[3]. 可数集合 S 上概率分布μ是一个 S 到[0,1]的映射函数,且满足 $\sum_{s\in S}\mu(s)=1$.

记号 $Dist(S)$表示集合 S 上概率分布的集合.记号$[s\mapsto 1]$表示 S 中的元素 s 上的点分布,即$[s\mapsto 1]\in Dist(S)$,且$[s\mapsto 1](s)=1$.在一项随机实验中,所有可能发生的结果形成的集合称为样本空间,记为Ω.Ω中的元素称为基本事件,Ω的子集称为事件,样本空间Ω称为必然事件,空集$\varnothing$称为不可能事件.在实际问题中,我们不是对所有的事件(样本空间Ω的所有子集)感兴趣,而是关心某些事件(Ω的某些子集)及其发生的概率,从而形成σ代数的概念.集合$\Pi\in 2^{\Omega}$称为Ω上的σ代数,当且仅当:

- $\Omega\in\Pi$;
- 如果 $E\in\Pi$,则$\Omega\backslash E\in\Pi$;
- 如果 $E_1,E_2,\ldots\in\Pi$,则$\bigcup_{i\geqslant 1}E_i\in\Pi$.

定义 2(概率空间)[3]. 概率空间是一个三元组 $PS=(\Omega,\Pi,\Pr)$.这里,Ω为样本空间;集合Π为Ω上的σ代数;$Pr:\Pi\to[0,1]$是度量函数,满足下面 3 个条件:

(1) 对任意的 $E\in\Pi$,$0\leqslant\Pr(E)\leqslant 1$;

(2) $Pr(\Omega)=1$;

(3) 对Π中两两不相交的事件 $E_1,E_2,\ldots$, $\Pr\left(\bigcup_{i=1}^{\infty}E_i\right)=\sum_{i=1}^{\infty}\Pr(E_i)$.

对于Π,称其中的任何元素都是可度量的.

2.2 概率时间自动机

本节主要回顾概率时间自动机的基本内容,并定义概率实时解释系统来描述多智体系统的动态行为.令$\mathbb{R}=[0,\infty)$为非负实数的集合,$\mathbb{N}=\{0,1,2,\ldots\}$为自然数的集合,$\chi$是有限时钟集,$v:\chi\to\mathbb{R}$为时钟赋值,$\mathbb{R}^{\chi}$是时钟集$\chi$上的赋值集合.

令 $t\in\mathbb{R}$,$v+t:\chi\to\mathbb{R}$形式化定义为$\forall x\in\chi,(v+t)(x)=v(x)+t$.

令 $X\subseteq\chi$,$v[X:=0]:\chi\to\mathbb{R}$形式化定义为$\forall x\in X,v[X:=0](x)=0,\forall x\in\chi\backslash X,v[X:=0](x)=v(x)$.

时钟约束ζ刻画了对时钟的约束,其形式化定义为

$$\zeta::=true|x\leqslant d|c\leqslant x|x+c\leqslant y+d|\neg\zeta|\zeta\vee\zeta|\zeta\wedge\zeta.$$

这里,$x,y\in\chi$,$c,d\in\mathbb{N}$.χ上时钟约束的集合定义为域,记为 $Zones(\chi)$.

时钟约束ζ和时钟赋值 v 的满足性关系$\models$定义如下:

- $v\models true$;
- $v\models x\leqslant d$ 当且仅当 $v(x)\leqslant d$;
- $v\models c\leqslant x$ 当且仅当 $c\leqslant v(x)$;
- $v\models x+c\leqslant y+d$ 当且仅当 $v(x)+c\leqslant v(y)+d$;
- $v\models\neg\zeta$当且仅当 $v\not\models\zeta$;
- $v\models\zeta_1\vee\zeta_2$ 当且仅当 $v\models\zeta_1$ 或者 $v\models\zeta_2$;
- $v\models\zeta_1\wedge\zeta_2$ 当且仅当 $v\models\zeta_1$ 且 $v\models\zeta_2$.

定义 3(概率时间自动机)[3]. 概率时间自动机 P 是一个八元组 $(L,\bar{l},\chi,Act,inv,enab,prob,\ell)$,其中,

- L 是有限的位置集合;
- $\bar{l}\in L$ 是初始位置;
- χ是时钟的集合;
- Act 是有限的动作集合;
- $Inv:L\to Zones(\chi)$是每个位置应该满足的不变量条件;
- $enab:L\times Act\to Zones(\chi)$是动作触发应该满足的条件;
- $prob:L\otimes Act\to 2^{Dist(2^{\chi}\times L)}$ 是概率转换函数;
- $\ell:L\to 2^{Ap}$ 是位置标记函数,其中,Ap 是原子命题集合.

在概率时间自动机中,时间的流失和动作的执行均可导致系统的状态发生变化.开始时,系统处于初始位置 $\bar{l}$,所有时钟的初始值为 0,并且以统一的速率增加.概率时间自动机上的状态是一个满足 $v\vDash inv(l)$的二元组 $(l,v)\in L\times\mathbb{R}^{\chi}$.在任何状态$(l,v)$下时间会自动流失,然后在某个时间点系统执行动作 $a\in Act$,从而导致系统所处的位置发生变化.时间的流失必须保证 $inv(l)$不能失效.对于动作 $a\in Act$,只有当 $v+t$ 满足 $enab(l,a)$时才能被选择执行.一旦执行动作 a,某个时钟集合将会被重置,后继的位置依据分布 $prob(l,a)$随机选择.

如图 2 所示为一个不可靠信道上的简单通信协议的概率时间自动机模型.系统由发送者和接受者两个主体组成,主体之间通过一个信道以及全局时钟 x,y 进行通信.消息的传递认为是瞬时的.初始状态设定为新的数据到达发送端,此时,x,y 的值设定为 0.发送者将数据保留 2~3 个时间单元,然后发送数据,同时将 x 的值设置为 0.数据成功到达接收端的概率是 0.90.接收端在接受到数据之后,必须在 1 个时间单元内给发送者发送消息到达的确认信息.确认信息成功到达发送端的概率是 0.95.一旦确认信息到达接收端,在下一个数据包到达发送端之前,时间就可以任意流失.但是,如果数据包丢失了,接受端将处于空闲状态,而发送端会一直处于等待确认信息的状态.因此,发送端将在发送信息之后的 2~3 个时间单元内重新发送消息.如果消息重发多于 7 个时间单元或者接受端接收到了数据,本次信息传递过程结束.

现在说明图 2 中原子命题表达的含义.发送端共有 4 个状态:接收到数据包、发送数据后等待数据确认信息、接收到确认信息、终止,分别用原子命题 *receivedata*,*waitack*,*receiveack*,*sabort* 表示.接受端共有 3 个状态:空闲、接收到数据、终止,分别用原子命题 *idle*,*receivedata*,*rabort* 表示.

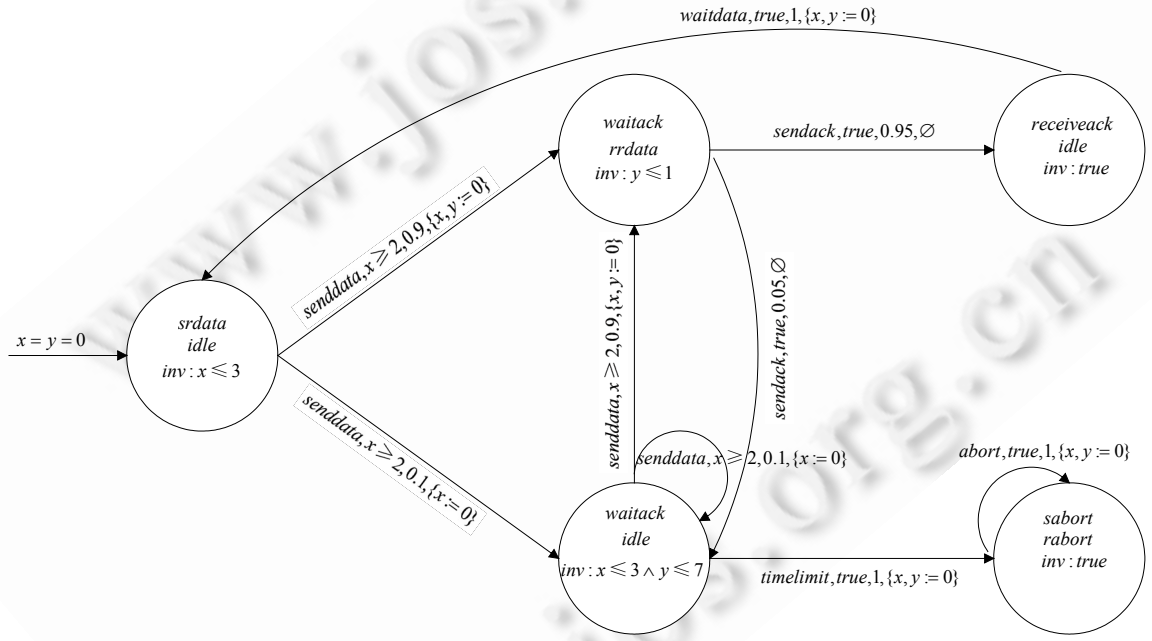

Fig.2 The probabilistic timed automata model of a simple communication protocol

图 2 一个简单通信协议的概率时间自动机模型

2.3 概率时间自动机的平行组合

对于分布 $p_1\in Dist(2^{\chi_1}\times L_1), p_2\in Dist(2^{\chi_2}\times L_2)$,定义分布 $p_1\otimes p_2\in Dist(2^{\chi_1\cup\chi_2}\times(L_1\times L_2))$ 如下:对于任意的 $X_1\subseteq\chi_1,X_2\subseteq\chi_2,l_1\in L_1,l_2\in L_2$,令 $p_1\otimes p_2(X_1\cup X_2,(l_1,l_2))=p_1(X_1,l_1)\cdot p_2(X_2,l_2)$.引入记号$(\varnothing,l_1)\mapsto 1$ 表示 $Dist(2^{\chi_1}\times L_1)$ 上的一种特殊概率分布:$(\varnothing,l_1)\mapsto 1((\varnothing,l_1))=1$,对任意的 $X_1\neq\varnothing\&X_1\subseteq\chi_1,l\in L_1,(\varnothing,l_1)\mapsto 1((X_1,l))=0$.引入记号$(\varnothing,l_2)\mapsto 1$ 表示 $Dist(2^{\chi_2}\times L_2)$ 上的一种概率分布:$(\varnothing,l_2)\mapsto 1((\varnothing,l_2))=1$,对任意的 $X_2\neq\varnothing\&X_2\subseteq\chi_2,l\in L_2,(\varnothing,l_2)\mapsto 1((X_2,l))=0$.

定义 4(概率时间自动机的平行组合)[3]. 令 $P_i=(L_i,\overline{l}_i,\chi_i,Act_i,inv_i,enab_i,prob_i,\ell_i)$ $(i\in\{1,2\})$为两个概率时间自动机,$\chi_1\cap\chi_2=\varnothing$.$P_1,P_2$ 的平行组合定义为

$$P_1\|P_2=(L_1\times L_2,(\overline{l}_1,\overline{l}_2),\chi_1\cup\chi_2,Act_1\cup Act_2,inv,enab,prob,\ell).$$

其中,

- 对任意的$(l_1,l_2)\in L_1\times L_2$,$inv(l_1,l_2)=inv_1(l_1)\wedge inv_2(l_2)$.
- $enab$:$(L_1\times L_2)\times(Act_1\cup Act_2)\to Zones(\chi_1\cup\chi_2)$定义如下:
  (1) $a\in Act_1\backslash Act_2$,$enab((l_1,l_2),a)=enab_1(l_1,a)$;
  (2) $a\in Act_2\backslash Act_1$,$enab((l_1,l_2),a)=enab_2(l_2,a)$;
  (3) $a\in Act_1\cap Act_2$,$enab((l_1,l_2),a)=enab_1(l_1,a)\wedge enab_2(l_2,a)$.
- $p\in prob((l_1,l_2),a)$当且仅当下面 3 个条件中的 1 个成立:
  (1) $a\in Act_1\backslash Act_2$,存在 $p_1\in prob(l_1,a)$,使得 $p=p_1\otimes\{(\varnothing,l_2)\mapsto 1\}$;
  (2) $a\in Act_2\backslash Act_1$,存在 $p_2\in prob(l_2,a)$,使得 $p=\{(\varnothing,l_1)\mapsto 1\}\otimes p_2$;
  (3) $a\in Act_1\cap Act_2$,存在 $p_1\in prob(l_1,a)$,$p_2\in prob(l_2,a)$,使得 $p=p_1\otimes p_2$.
- $\ell(l_1,l_2)=\ell(l_1)\cup\ell(l_2)$.

图 3、图 4 分别给出了两个简单的概率时间自动机 P_1,P_2,现在考察 P_1 与 P_2 的平行组合 $P_1\|P_2$.对于初始位置(L_{11},L_{21}):在 P_1 中,初始位置 L_{11} 下,a 是唯一可以被触发的动作;在 P_2 中,初始位置 L_{21} 下,b 是唯一可以被触发的动作.因为 $a\in Act_1\backslash Act_2$,所以 a 的执行会导致 P_1 的状态发生变化,P_2 的状态保持不变.对于动作 b,因为 $a\in Act_2\backslash Act_1$,所以 b 的执行会导致 P_2 中的状态发生变化,P_1 中的状态保持不变.在 P_1 中,a 的执行导致位置变为 L_{12} 的概率是 0.6,变为 L_{13} 的概率是 0.4,因此在 $P_1\|P_2$ 中,执行动作 a 导致(L_{11},L_{21})变为(L_{12},L_{21})的概率是 0.6,变为(L_{13},L_{21})的概率是 0.4.在 P_2 中,b 的执行导致位置变为 L_{21} 的概率是 0.2,变为 L_{22} 的概率是 0.8,因此在 $P_1\|P_2$ 中,执行动作 b 导致(L_{11},L_{21})变为(L_{11},L_{21})的概率是 0.2,导致变为(L_{11},L_{22})的概率是 0.8.

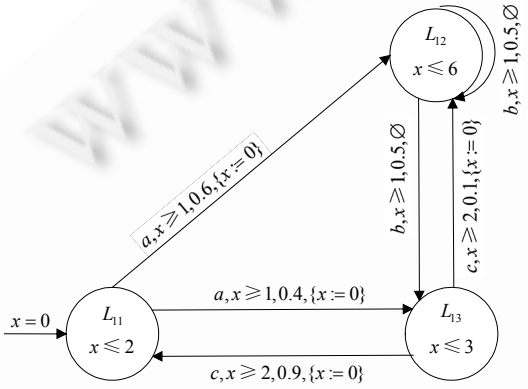

Fig.3 The probabilistic timed automata P_1

图 3 概率时间自动机 P_1

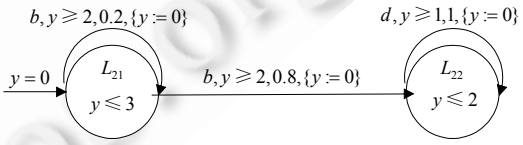

Fig.4 The probabilistic timed automata P_2

图 4 概率时间自动机 P_2

上述考察的动作 a 和 b 在(L_{11},L_{21})下执行只能导致 1 个位置发生变化,而对于位置(L_{12},L_{21}),执行动作 b 会导致两个位置同时发生变化.在 L_{12} 下,执行 b 导致位置从 L_{12} 变为 L_{13} 的概率是 0.5;在 L_{21} 下,执行 b 导致位置从 L_{21} 变为 L_{22} 的概率是 0.8.因此在(L_{12},L_{21})下,执行 b 导致位置变为(L_{13},L_{22})的概率是 0.4;同样地,变为(L_{13},L_{22})的概率是 0.4,变为(L_{13},L_{21})的概率是 0.1,位置不发生改变的概率是 0.1.依据定义 4,最终的平行组合 $P_1\|P_2$ 如图 5 所示.

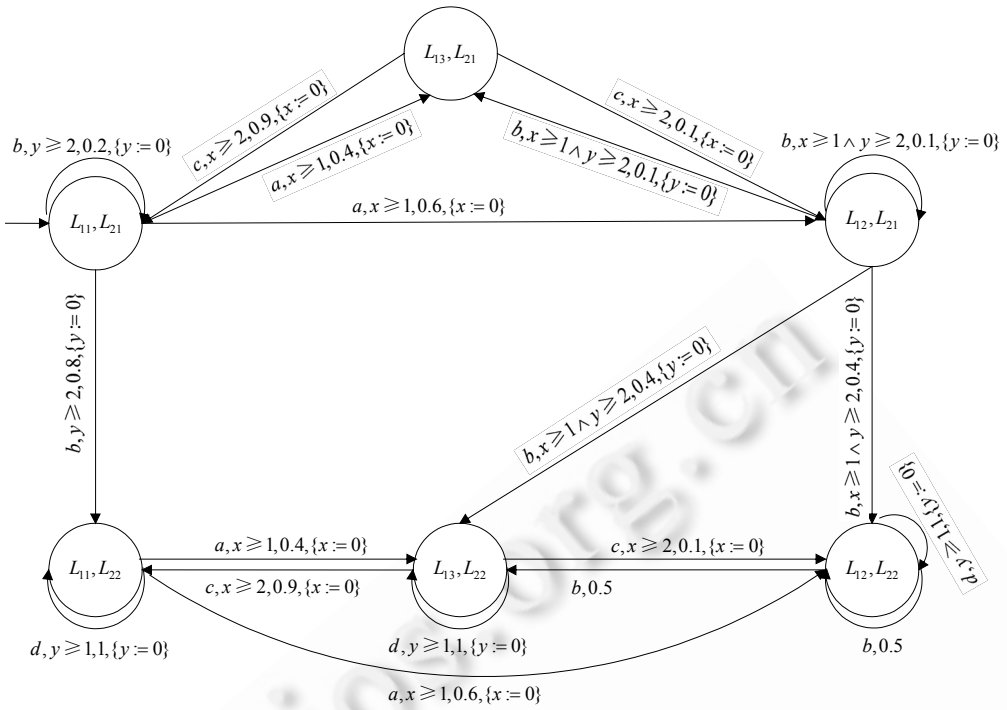

Fig.5 The parallel composition $P_1||P_2$ of probabilistic timed automata P_1, P_2

图 5 概率时间自动机 P_1,P_2 的平行组合 $P_1||P_2$

2.4 概率时间自动机的语义

定义 5[3]. 一个马尔可夫决策过程 M 是一个五元组 $(S,\overline{s},Action,Step,Label)$,其中,

- S 是状态集;
- $\overline{s}\in S$ 是初始状态;
- $Action$ 是动作集;
- $Step$:$S\times Act\to 2^{Dist(S)}$是概率转换函数;
- $Label$:$S\to 2^{Ap}$ 是状态标记函数,其中,Ap 是原子命题集合.

如图 6 所示为一个简单的马尔可夫决策过程.在该过程中,$S=\{\overline{s},s_1,s_2\}$,$\overline{s}$ 是初始状态,$Action=\{a,b,c\}$,概率转换函数为 $Step(\overline{s},c)(\overline{s})=\frac{1}{2}$,$Step(\overline{s},c)(s_2)=\frac{1}{2}$,$Step(\overline{s},b)(s_1)=1$,$Step(s_2,a)(\overline{s})=1$,$Step(s_1,a)(\overline{s})=1$,状态标记函数为 $Label(\overline{s})=\{q\}$,$Label(s_1)=\{p\}$,$Label(s_2)=\{r\}$.

在马尔可夫决策过程 M 中,路径是一条有穷的序列 $s_0,a_0,s_1,a_1,s_2,a_2,\ldots,a_{n-1},s_n$,满足:

$$\forall 0\leqslant i\leqslant n-1,Step(s_i,a_i)(s_{i+1})>0.$$

称状态 s 是从状态 s_0 可达的当且仅当存在路径 $s_0,a_0,s_1,a_1,s_2,a_2,\ldots,a_{n-1},s_n$,使得 $s_n=s$.

定义 6(概率时间自动机的语义)[3]. 令 $P=(L,\overline{l},\chi,Act,inv,enab,prob,\ell)$ 是一个概率时间自动机,其语义定义为一个马尔可夫决策过程 $(S,\overline{s},Action,Step,Label)$,其中,

- $S=\{(l,v)\in L\times\mathbb{R}^{\chi}|v\vDash inv(l)\}$;
- $\overline{s}=(\overline{l},0)$;
- $Action=\mathbb{R}\times Act$;
- $\lambda\in Step((l,v),(t,a))$当且仅当对于任意的 $0\leqslant t'\leqslant t,v+t'\vDash inv(l),v+t\vDash enab(l,a)$,对任意的$(l',v')\in S$,

$$\lambda(l',v')=\sum\{prob(l,a)(X,l')\mid X\in 2^{\chi}\wedge v'=(v+t)[X:=0]\};$$

- $Label(l,v)=\ell(l)$.

考察图 4 中概率时间自动机 P_2,设马尔可夫决策过程 M_2 是 P_2 的语义,如图 7 所示.在 M_2 中,初始状态是 $(L_{21},0)$.由 P_2 中 L_{21} 下动作 b 触发的条件可知,在 M_2 中,动作$(1.5,b)$不会导致初始状态发生任何变化,而动作$(2.1,b)$的触发会导致状态发生变化,具体变化是以 0.2 的概率变为状态$(L_{21},0)$,0.8 的概率变为状态$(L_{22},0)$.

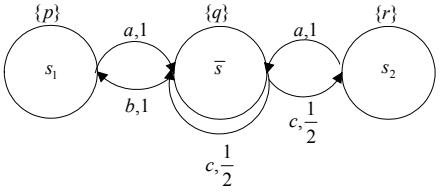

Fig.6 A simple Markov decision process

图 6 一个简单的马尔可夫决策过程

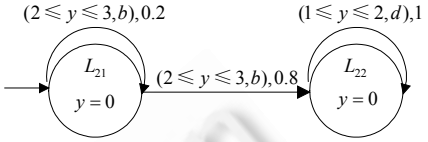

Fig.7 The semantics of P_2

图 7 P_2 的语义解释

2.5 概率实时解释系统

令 $P=(L,\bar{l},\chi,Act,inv,enab,prob,\ell)$ 为概率时间自动机,定义 $Zones_\chi(P)$为 P 中所有出现在不变量约束 inv 和触发条件 $enab$ 中时钟约束的集合.形式化定义为 $Zones_\chi(P)=\{inv(l)\in Zones(\chi)|l\in L\}\cup\{enab(l,a)|l\in L,a\in Act\}$.定义 $c_{\max}(P)=\max\{c_{\max}(\zeta)|\zeta\in Zones_\chi(P)\}$,其中,$c_{\max}(\zeta)$表示约束$\zeta$中出现的最大常量.对于任意的实数 r,记号$\lfloor r\rfloor$表示 r 的整数部分,$frac(r)$表示 r 的小数部分.

定义 7(时钟赋值等价)[13]. 令 $P=(L,\bar{l},\chi,Act,inv,enab,prob,\ell)$ 为概率时间自动机,对任意的时钟赋值 $v,v'\in\mathbb{R}^\chi$,$v,\simeq v'$当且仅当下面的条件得到满足:

1. 对任意的 $x\in\chi$,$v(x)>c_{\max}(P)$当且仅当 $v'(x)>c_{\max}(P)$;
2. 对任意的 $x,y\in\chi$,如果 $v(x)\leqslant c_{\max}(P)$,$v'(x)\leqslant c_{\max}(P)$,那么,
   (a) $\lfloor v(x)\rfloor=\lfloor v'(x)\rfloor$;
   (b) $frac(v(x))=0$ 当且仅当 $frac(v'(x))=0$;
   (c) $frac(v(x))\leqslant frac(v(y))$当且仅当 $frac(v'(x))\leqslant frac(v'(y))$.

直觉上,两个时钟赋值是等价的当且仅当:1) 对同样的时钟,它们的值同时都大于 $c_{\max}(P)$;或者 2) 整数部分相同,小数部分为 0 或者保持序的关系.等价关系将时钟赋值划分成了不同的等价类.如图 8 所示为当$\chi=\{x,y\}$,$c_{\max}(P)=2$ 时,时钟赋值等价关系的划分.

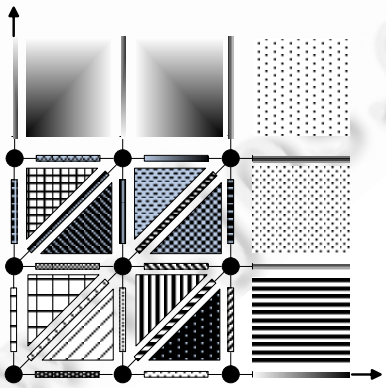

Fig.8 Equivalence class partition of two clock variables when $c_{\max}(P)=2$

图 8 $c_{\max}(P)=2$ 的情形下两个时钟变量的等价类划分

令 $Ag=\{1,\ldots,n\}$表示智体的集合,每个智体的行为模型化为一个概率时间自动机:

$$P_i=(L_i,\overline{l}_i,\chi_i,Act_i,inv_i,enab_i,prob_i,\ell_i).$$

令 $P=(L,\overline{l},\chi,Act,inv,enab,prob,\ell)$ 为 $P_1,\ldots,P_n$ 的平行组合,用来表示 n 智体组成的系统的行为.首先定义位置函数 Loc_i:对于组合系统中的任一位置 $l=(l_1,\ldots,l_n)$,$Loc_i(l)=l_i$.

定义 8(概率实时解释系统). 一个概率实时解释系统 PM 是一个多元组:

$$(S,\overline{s},\mathbb{R}\times Act,Step,Label,\sim_1,\ldots,\sim_n,PK_1,\ldots,PK_n),$$

其中,

- S 是 $L\times\mathbb{R}^x$的子集,且从初始状态 $(\overline{l},0)$ 可达;
- $(S,\overline{s},\mathbb{R}\times Act,Step,Label)$ 是一个马尔可夫决策过程;
- $\sim_i\subseteq S\times S$ 是认知等价关系:$(l,v)\sim_i(l',v')$当且仅当 $Loc_i(l)=Loc_i(l'),v=v'$;
- $PK_i:S\to Dist(S)(1\leqslant i\leqslant n)$是概率认知关系,满足 $\sum_{s\sim_i s'}PK_i(s)(s')=1$,且对任意的$(l,v),(l',v'),(l_1,v_1),(l_1',v_1')$,如果 $(l,v)\sim_i(l_1,v_1),(l',v')\sim_i(l_1',v_1')$,则 $PK_i(l,v)(l',v')=PK_i(l_1,v_1)(l_1',v_1')$.

概率实时解释系统 PM 上的路径π是一条非空的有穷或者无穷序列: $s_0\xrightarrow{(t_0,a_0),p_0}s_1\xrightarrow{(t_1,a_1),p_1}s_2\to\ldots$,其中,对任意的 $i\geqslant 0,s_i\in S,p_i\in Step(s_i,(t_i,a_i)),p_i(s_{i+1})>0$.

为便于表述,引入下列记号:

- $\pi_i=s_0\xrightarrow{(t_0,a_0),p_0}s_1\xrightarrow{(t_1,a_1),p_1}s_2\to\ldots\xrightarrow{(t_{i-1},a_{i-1}),p_{i-1}}s_i$:π的前缀;
- $\pi(i)=s_i$:π上的第 i 个状态;
- $|\pi|$:π的长度,即转换关系的数目(无穷路径长度为∞);
- $Path_{fin}$:有穷路径的集合;
- $Path_{fin}(s)$:从 s 出发的有穷路径的集合;
- $Path_{ful}$:无穷路径的集合;
- $Path_{ful}(s)$:从 s 出发的无穷路径的集合.

考察无穷路径π,称二元组(i,t')是一个方位当且仅当 $i\leqslant|\pi|,t'\in\mathbb{R},0\leqslant t'\leqslant t_i$.方位$(i,t')$处的状态表示为 s_i+t':与 s_i 具有相同的位置,时钟赋值为 s_i 中的时钟赋值加上 t'.给定路径π,称方位(j,t')是(i,t)的前驱,记为$(j,t')\prec(i,t)$,当且仅当 $j<i$ 或者 $j=i,t'<t$.

定义 9(路径延迟). 对任意的路径π,任意的自然数 $0\leqslant i\leqslant|\pi|$,定义 $D_\pi(i)$表示直到第 i 个转换发生所流失的时间.形式化定义为:$D_\pi(0)=0$;对任意的 $0\leqslant i\leqslant|\pi|$, $D_\pi(i)=\sum_{j=0}^{i-1}t_j$.

进一步来说,无穷路径π是发散的当且仅当对任意的 $t\in\mathbb{R}$,存在 $i\in\mathbb{N}$,使得 $D_\pi(i)>t$.现在引入调度的概念,调度的主要目的是解决模型中的非确定性选择问题.

定义 10(调度). 概率实时解释系统 $PM=(S,\overline{s},\mathbb{R}\times Act,Step,Label,\sim_1,\ldots,\sim_n,PK_1,\ldots,PK_n)$ 上的调度θ是映射有穷路径 $\pi=s_0\xrightarrow{(t_0,a_0),p_0}s_1\xrightarrow{(t_1,a_1),p_1}s_2\to\ldots\xrightarrow{(t_{m-1},a_{m-1}),p_{m-1}}s_m$ 到概率分布 $Dist(S)$的函数,且满足:

$$\theta(\pi)\in Step(s_m,(t_m,a_m)).$$

令Θ表示 PM 上所有调度的集合.

对于调度θ,定义如下记号:

- $Path_{fin}^{\theta}$:满足 $p_i=\theta(\pi_i)$的有穷路径的集合;
- $Path_{ful}^{\theta}$:满足 $p_i=\theta(\pi_i)$的无穷路径的集合.

对任意的概率实时解释系统 PM、调度θ,定义Π^θ为 $Path_{ful}^{\theta}$ 上包含 $\bigcup_{\pi_i\in Path_{fin}^{\theta}}\{\pi\mid\pi\in Path_{ful}^{\theta}\ \&\ \pi_i\text{是}\pi\text{的前缀}\}$ 的最小σ代数.

定义 $Path_{fin}^{\theta}\to[0,1]$ 上的概率计算函数 Pr_{fin}^{θ}:

- 如果$|\pi|=0$,则 $Pr^{\theta}_{fin}(\pi)=1$;
- 对任意有穷路径 $\pi'\in Path^{\theta}_{fin}$,如果 $\pi'=\pi\xrightarrow{(t,a),p}s$,则 $Pr^{\theta}_{fin}(\pi')=Pr^{\theta}_{fin}(\pi)\cdot p(s)$.

定义 11(概率度量函数). 概率度量函数 Pr^{θ}定义为 $Pr^{\theta}\{\pi\mid\pi\in Path^{\theta}_{ful}\ \&\ \pi_i\in Path^{\theta}_{fin}\}=Pr^{\theta}_{fin}(\pi_i)$.

3 概率实时认知逻辑 PTCTLK

为了推理多智体系统的行为属性,我们在 TCTLK 上引入概率度量算子,提出一种概率实时认知逻辑 PTCTLK.该逻辑可以对概率、实时性以及知识进行推理.

3.1 语 法

令 Ag 为原子命题集,$Ag=\{1,\ldots,n\}$表示智体的集合,γ为公式中出现的时钟集合,简称为公式时钟集.

定义 12(PTCTLK 的语法法). PTCTLK 的语法定义如下:

$$\phi::=true\mid a\mid\zeta\mid\phi\wedge\phi\mid\neg\phi\mid z.\phi\mid[\phi\exists U\phi]_{\triangleright\eta}\mid[\phi\forall U\phi]_{\triangleright\eta}\mid[\exists G\phi]_{\triangleright\eta}\mid[\forall G\phi]_{\triangleright\eta}\mid[\phi\exists R\phi]_{\triangleright\eta}\mid[\phi\forall R\phi]_{\triangleright\eta}\mid[K_i\phi]_{\triangleright\eta}\mid[E_\Gamma\phi]_{\triangleright\eta}.$$

其中,$a\in Ap$ 是原子命题,$\zeta\in Zones(\chi\cup\gamma)$是时钟约束,$z\in\gamma,\triangleright\in\{>,\geqslant\}$,$\Gamma\subseteq Ag$.

利用 PTCTLK 可以表示以下属性:

- 在任何调度下,系统在 5 个时间单元内发送第 1 个数据包而没有发送第 2 个数据包的概率大于 0.99:
$$z.[packet2unsent\forall U(packet1delivered\wedge(z<5))]_{>0.99}.$$
- 在任何调度下,在 8 个时间单元流失之前,系统时钟 x 的值不超过 3 的概率不低于 0.95:
$$z.[(x\leqslant 3)\forall U(z=8)]_{\geqslant 0.95}.$$

3.2 语 义

令 $\Im:\gamma\to\mathbb{R}$表示公式时钟赋值.注意到,χ和γ中时钟的值可分别从状态和公式时钟赋值获得.给定一个状态 $s=(l,v)$,公式时钟赋值$\Im$,时钟约束$\zeta\in Zones(\chi\cup\gamma)$,定义记号$\zeta[s,\Im]$表示将$\zeta$中任意系统时钟 $x\in\chi$的每一次出现,任意公式时钟 $z\in\gamma$的每一次出现分别替换为 $v(x)$,$\Im(z)$获得的布尔函数值.$\zeta[s,\Im]=true$ 当且仅当$(v,\Im)\vDash\zeta$.

定义 13(PTCTLK 语义). 给定一个概率实时解释系统

$$PM=(S,\bar{s},\mathbb{R}\times Act,Step,Label,\sim_1,\ldots,\sim_n,PK_1,\ldots,PK_n).$$

PM 上的调度集合Θ,对 PM 中的任意状态 s,时钟赋值$\Im$,PTCTLK 公式ϕ,满足性关系 $s,\Im\vDash_\Theta\phi$递归定义如下:

- $s,\Im\vDash_\Theta true$ 对所有的 s 和$\Im$都成立.
- $s,\Im\vDash_\Theta a$ 当且仅当 $a\in Label(s)$.
- $s,\Im\vDash_\Theta\zeta$当且仅当$\zeta[s,\Im]=true$.
- $s,\Im\vDash_\Theta\phi_1\wedge\phi_2$ 当且仅当 $s,\Im\vDash_\Theta\phi_1$ 且 $s,\Im\vDash_\Theta\phi_2$.
- $s,\Im\vDash_\Theta z.\phi$当且仅当 $s,\Im[z:=0]\vDash_\Theta\phi$.
- $s,\Im\models_\Theta[\phi_1\exists U\phi_2]_{\triangleright\eta}$ 当且仅当存在调度$\theta\in\Theta$,使得 $Pr^{\theta}(\{\pi\mid\pi\in Path^{\theta}_{ful}(s)\ \&\ \pi,\Im\models_\Theta\phi_1U\phi_2\})\triangleright\eta$.
- $s,\Im\models_\Theta[\phi_1\forall U\phi_2]_{\triangleright\eta}$ 当且仅当对任意的调度$\theta\in\Theta$, $Pr^{\theta}(\{\pi\mid\pi\in Path^{\theta}_{ful}(s)\ \&\ \pi,\Im\models_\Theta\phi_1U\phi_2\})\triangleright\eta$.
- $\pi,\Im\vDash_\Theta\phi_1U\phi_2$ 当且仅当存在位置(j,t),使得$\pi(j)+t,\Im+D_\pi(j)+t\vDash_\Theta\phi_2$,对所有的位置$(j',t')$,如果$(j',t')<(j,t)$,则
$$\pi(j')+t',\Im+D_\pi(j')+t'\vDash_\Theta\phi_1.$$
- $s,\Im\models_\Theta[\exists G\phi]_{\triangleright\eta}$ 当且仅当存在调度$\theta\in\Theta$,使得 $Pr^{\theta}(\{\pi\mid\pi\in Path^{\theta}_{ful}(s)\ \&\ \pi,\Im\models_\Theta G\phi\})\triangleright\eta$.
- $s,\Im\models_\Theta[\forall G\phi]_{\triangleright\eta}$ 当且仅当对任意的调度$\theta\in\Theta$, $Pr^{\theta}(\{\pi\mid\pi\in Path^{\theta}_{ful}(s)\ \&\ \pi,\Im\models_\Theta G\phi\})\triangleright\eta$.
- $\pi,\Im\vDash_\Theta G\phi$当且仅当对任意的位置$(j,t)$,$\pi(j)+t,\Im+D_\pi(j)+t\vDash_\Theta\phi$.
- $s,\Im\models_\Theta[\phi_1\exists R\phi_2]_{\triangleright\eta}$ 当且仅当存在调度$\theta\in\Theta$,使得 $Pr^{\theta}(\{\pi\mid\pi\in Path^{\theta}_{ful}(s)\ \&\ \pi,\Im\models_\Theta\phi_1R\phi_2\})\triangleright\eta$.
- $s,\Im\models_\Theta[\phi_1\forall R\phi_2]_{\triangleright\eta}$ 当且仅当对任意的调度$\theta\in\Theta$, $Pr^{\theta}(\{\pi\mid\pi\in Path^{\theta}_{ful}(s)\ \&\ \pi,\Im\models_\Theta\phi_1R\phi_2\})\triangleright\eta$.
- $\pi,\Im\vDash_\Theta\phi_1R\phi_2$ 当且仅当:1) 存在位置(j,t),使得$\pi(j)+t,\Im+D_\pi(j)+t\vDash_\Theta\phi_1$,对所有的位置$(j',t')$,如果$(j',t')<(j,t)$或

者$(j',t')=(j,t)$,则$\pi(j')+t',\Im+D_\pi(j')+t'\models_\Theta\phi_2$;或者 2) 对任意的位置$(j,t),\pi(j)+t,\Im+D_\pi(j)+t\models_\Theta\phi_2$.

- $s,\Im\models_\Theta[K_i\phi]_{\rhd\eta}$ 当且仅当 $\sum_{s',\Im\models_\Theta\phi}PK_i(s,s')\rhd\eta$.
- $s,\Im\models_\Theta[E_\Gamma\phi]_{\rhd\eta}$ 当且仅当对任意的 $i\in\Gamma,s,\Im\models_\Theta[K_i\phi]_{\rhd\eta}$,即 $\sum_{s',\Im\models_\Theta\phi}PK_i(s,s')\rhd\eta$.

概率时态逻辑 PCTL 是在 CTL 的基础上引入概率度量算子而得到的一种逻辑系统,且两者是不等价的,CTL 也不是 PCTL 的子集.因为 CTL 是实时认知逻辑 TCTLK 的子集,PCTL 是 PTCTLK 的子集,所以 PTCTLK 与 TCTLK 不等价,且 TCTLK 也不是 PTCTLK 的子集.

4 概率知识区域图

概率时间自动机的模型检测是通过区域图实现的.这一节我们首先将时钟赋值等价扩展到系统时钟χ和公式时钟γ上.令 $c_{\max}(\phi)$表示公式ϕ中出现的最大常量.

定义 14(扩展时钟赋值等价). 令 $P=(L,\bar{l},\chi,Act,inv,enab,prob,\ell)$ 是一个概率时间自动机,ϕ为 PTCTLK 公式,对任意的时钟赋值 $v,v'\in\mathbb{R}^{\chi\cup\gamma},v,\simeq v'$当且仅当下面的条件得到满足:

1. 对任意的 $x\in\chi\cup\gamma,v(x)>\max(c_{\max}(P),c_{\max}(\phi))$当且仅当 $v'(x)>\max(c_{\max}(P),c_{\max}(\phi))$.
2. 对任意的 $x,y\in\chi\cup\gamma$,如果 $v(x)\leqslant\max(c_{\max}(P),c_{\max}(\phi)),v'(x)\leqslant\max(c_{\max}(P),c_{\max}(\phi))$,那么,
   (a) $\lfloor v(x)\rfloor=\lfloor v'(x)\rfloor$;
   (b) $frac(v(x))=0$ 当且仅当 $frac(v'(x))=0$;
   (c) $frac(v(x))\leqslant frac(v(y))$当且仅当 $frac(v'(x))\leqslant frac(v'(y))$.

引入记号$[v]$表示与 v 等价的时钟赋值形成的等价类.给定概率时间自动机 P 和 PTCTLK 公式ϕ,记号 $equiv(P,\phi)$表示时钟赋值等价类的集合.

定义 15(时钟等价类的满足性). 给定概率时间自动机 P 和 PTCTLK 公式ϕ,令$\alpha\in equiv(P,\phi),\zeta$是一个时钟约束.称$\alpha$满足$\zeta$当且仅当对任意的$(v,\Im)\in\alpha,\zeta[s,\Im]$为真.

定义 16(区域的刻画). 令$\alpha,\beta\in equiv(P,\phi)$为$\mathbb{R}^{\chi\cup\gamma}$上不同的等价类:

1. 后继:β为α的后继,记为 $succ(\alpha)$,当且仅当对每个$(v,\Im)\in\alpha$,存在一个正实数 $t\in\mathbb{R}$,使得$(v+t,\Im+t)\in\beta$,且对任意的 $t'\leqslant t,(v+t',\Im+t')\in\alpha\cup\beta$.
2. x 零类:对$\chi\cup\gamma$中的任意时钟 x,称等价类α为 x 零类当且仅当对任意的$(v,\Im)\in\alpha,(v,\Im)(x)=0$.
3. x 无界类:对$\chi\cup\gamma$中的任意时钟 x,称等价类α为 x 无界类当且仅当对任意的$(v,\Im)\in\alpha$,
$$(v,\Im)(x)>\max(c_{\max}(P),c_{\max}(\phi)).$$

考察如下实例:令 x 是系统时钟,z 是公式时钟,$\max(c_{\max}(P),c_{\max}(\phi))=1$,等价类之间的后继关系如图 9 所示.

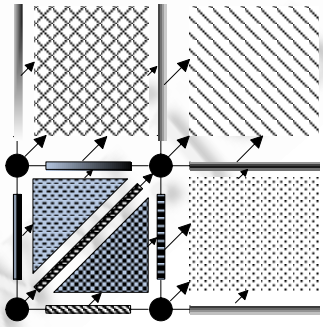

Fig.9 Subsequence relation between equivalence classes

图 9 等价类之间的后继关系

图9中,箭头指向的等价类是箭头发起端代表的等价类的后继(忽略了自身到自身的后继关系).由图9可知,等价类[(0,0.5)]是 x 零类,等价类[(1.5,1.5)]是 x 无界类.

后继关系可自然地扩展到状态上面:如果 $l'=l,\beta=succ(\alpha)$,则称(l',β)是(l,α)的后继.类似地,如果α为 x 零类,则(l,α)也是 x 零类;如果α为 x 无界类,则(l,α)也是 x 无界类.

定义 17(概率知识区域图). 令 $P=(L,\bar{l},\chi,Act,inv,enab,prob,\ell)$ 为一概率时间自动机,ϕ为 PTCTLK 公式,$PM=(S,\bar{s},\mathbb{R}\times Act,Step,Label,\sim_1,...,\sim_n,PK_1,...,PK_n)$ 为与 P 对应的概率实时解释系统.概率知识区域图 $Region(P,\phi)$是一个多元组 $(S^*,\bar{s}^*,Act^*,Step^*,Label^*,PK_1^*,...,PK_n^*)$:

- $S^*=L\times equiv(P,\phi)$为区域的集合.
- $\bar{s}^*=(\bar{l},(0,0))$ 是初始状态.
- $Act^*=Act$ 是动作集.
- $Step^*:S^*\to 2^{Dist(s^*)}$ 定义如下:

  (1) 时间流逝,无动作发生:如果 $succ(\alpha)$满足 $inv(l)$,那么 $p^{l,\alpha}\in Step^*((l,\alpha))$.这里,对任何的$(l',\beta)\in S^*$,

$$p^{l,\alpha}=\begin{cases}1, & \text{if } (l',\beta)=(l,succ(\alpha))\\ 0, & \text{otherwise}\end{cases}.$$

  (2) 离散转换:$p^{l,\alpha}\in Step^*((l,\alpha))$当且仅当存在动作 $a\in Act,p'\in prob(l,a)$,使得α满足触发条件 $enab(l,\alpha)$,任意的 l'和等价类β, $p^{l,\alpha}(l',\beta)=\sum_{X\subseteq\chi\&\alpha[X:=0]=\beta}p'(l',X)$.

- $Label^*:S\to 2^{Ap}$ 是状态标记函数,其中,$Label^*(l,\alpha)=\ell(l)$.
- $PK_i^*:S^*\to Dist(S^*)$ 定义为:设 $s^*=(l,\alpha),s_1^*=(l',\beta),v\in\alpha,v'\in\beta,PK_i^*(s^*)(s_1^*)=PK_i(l,v)(l',v')$.

由 PK_i 的定义可知,如果 $(l,v)\sim_i(l_1,v_1),(l',v')\sim_i(l_1',v_1')$,则 $PK_i(l,v)(l',v')=PK_i(l_1,v_1)(l_1',v_1')$.因此, PK_i^* 是良定义的,即与 v,v'的选择无关.

考察如图 10 所示的概率时间自动机 P_3.令 x 是系统时钟,z 是公式时钟,ϕ是 PTCTLK 公式,

$$\max(c_{\max}(P_3),c_{\max}(\phi))=1.$$

图 11 给每个等价类进行了标号.考察初始状态 $\bar{s}^*=(\bar{l},\alpha_1)$ 下执行各类动作引起的状态的转换.首先考虑时间流逝,在这种情况下,位置不会发生变化,时钟 x,z 同步增长,比如变为(0.5,0.5).此时,(0.5,0.5)属于等价类α_3,因此 $\bar{s}^*=(\bar{l},\alpha_1)$ 转换为状态 $(\bar{l},\alpha_3)$ 的概率为 1.在 $\bar{s}^*$ 下执行动作 b,位置保持不变的概率是 0.2,变为 l_1的概率是 0.8.因此执行动作 b,状态 $(\bar{l},\alpha_1)$ 保持不变的概率是 0.2,变为状态 $(\bar{l},\alpha_1)$ 的概率是 0.8.在 $\bar{s}^*$ 下执行动作 a,位置保持不变的概率是 1,因此执行动作 a,状态 $(\bar{l},\alpha_1)$ 保持不变的概率是 1.

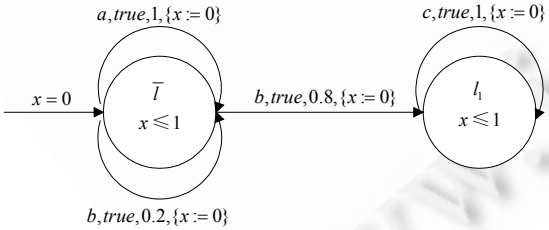

Fig.10 Probabilistic timed automata P_3

图 10 概率时间自动机 P_3

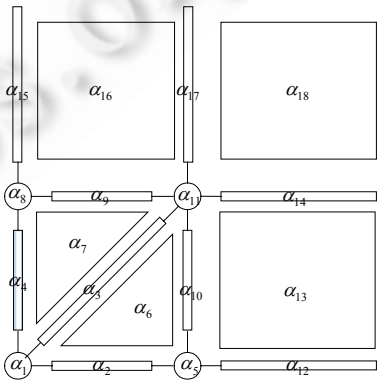

Fig.11 Equivalence class of clock assignment

图 11 时钟赋值等价类

现在考察状态 $(\overline{l},\alpha_{16})$.首先,在$\alpha_{16}$中选择元素$(x,z)$=(0.5,1.5).当 x=0.5 时,依据概率时间自动机 P_3 的语义解释,动作 a 和 b 都是可以执行的.当执行动作 a 时,位置保持不变的概率是 1,且 x 被重置,因此状态变为 $(\overline{l},(0,1.5))$.而$(0,1.5)\in\alpha_{15}$,因此执行动作 a,状态将由 $(\overline{l},\alpha_{16})$ 变为 $(\overline{l},\alpha_{15})$,且概率为 1.同理,当执行动作 b 时,位置保持不变的概率是 0.2,位置变为 l_1 的概率是 0.8,且 x 被重置.在位置保持不变的情况下,状态变为 $(\overline{l},(0,1.5))$,而$(0,1.5)\in\alpha_{15}$,因此状态变为 $(\overline{l},\alpha_{15})$ 的概率为 1.位置变为 l_1 时,状态变为$(l_1,(0,1.5))$,而$(0,1.5)\in\alpha_{15}$,因此状态变为(l_1,α_{15})的概率为 0.8.最终的概率知识区域如图 12 所示.

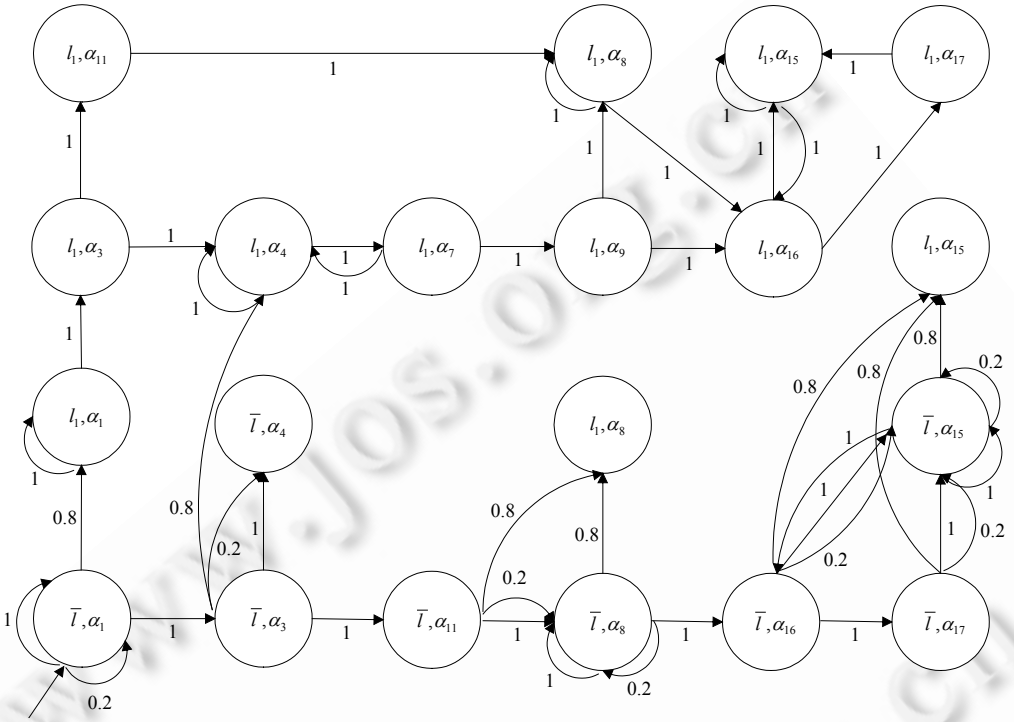

Fig.12 Probabilistic knowledge region of P_3

图 12 P_3 的概率知识区域图

定义 18(概率知识区域图上的路径). $Region(P,\phi)=(S^*,\overline{s}^*,Act^*,Step^*,Label^*,PK_1^*,...,PK_n^*)$ 上的路径是一个无穷或者有穷的序列,具体形式如下:

$$\pi^*=(l_0,\alpha_0)\xrightarrow{\mu_0^*}(l_1,\alpha_1)\xrightarrow{\mu_1^*}...,$$

其中,$l_i\in L,\alpha_i\in equiv(P,\phi)$, $\mu_i^*\in Step^*(l_i,\alpha_i)$,且满足 $\mu_i^*(l_{i+1},\alpha_{i+1})>0$.

定义 19(概率知识区域图上的调度). $Region(P,\phi)=(S^*,\overline{s}^*,Act^*,Step^*,Label^*,PK_1^*,...,PK_n^*)$ 上的调度θ^*是有穷路径 $\pi^*=(l_0,\alpha_0)\xrightarrow{\mu_0}(l_1,\alpha_1)\xrightarrow{\mu_1}...\xrightarrow{\mu_{m-1}}(l_m,\alpha_m)$ 到概率分布 $Dist(S^*)$的映射函数,满足:

$$\theta^*(\pi^*)\in Step^*(s^*,(l_m,\alpha_m)).$$

令Θ^*表示所有调度的集合.

对于调度$\theta^*\in\Theta^*$,定义如下记号:

- $Path_{fin}^{\theta^*}$:满足 $\mu_i^*=\theta^*(\pi_i^*)$ 的有穷路径的集合.
- $Path_{ful}^{\theta^*}$:满足 $\mu_i^*=\theta^*(\pi_i^*)$ 的无穷路径的集合.
- $Path_{fin}^{\theta^*}(s^*)$:从 s^*出发满足 $\mu_i^*=\theta^*(\pi_i^*)$ 的有穷路径的集合.

- $Path_{ful}^{\theta^*}(s^*)$:从 s^*出发满足 $\mu_i^* = \theta^*(\pi_i^*)$ 的无穷路径的集合.

对任意的概率知识区域图 $Region(P,\phi)=(S^*,\bar{s}^*,Act^*,Step^*,Label^*,PK_1^*,...,PK_n^*)$ 上的调度 θ^*,定义 Π^{θ^*} 为 $Path_{ful}^{\theta^*}$ 上包含 $\bigcup_{\pi_i \in Path_{fin}^{\theta^*}} \{\pi \mid \pi \in Path_{ful}^{\theta^*} \,\&\, \pi_i$是$\pi$的前缀$\}$ 的最小σ代数.

定义 $Path_{fin}^{\theta^*} \to [0,1]$ 上的概率计算函数 $Pr_{fin}^{\theta^*}$:

- 如果$|\pi^*|$=0,则 $Pr_{fin}^{\theta^*}(\pi^*)=1$;
- 对任意有穷路径 $\pi'^* \in Path_{fin}^{\theta^*}$, 如果 $\pi'^* = \pi^* \xrightarrow{\mu^*} s^*$, 则 $Pr_{fin}^{\theta^*}(\pi'^*) = Pr_{fin}^{\theta^*}(\pi^*)\cdot\mu^*(s^*)$.

定义 20(概率度量函数). 概率度量函数 Pr^{θ^*} 定义为 $Pr^{\theta^*}\{\pi^* \mid \pi^* \in Path_{ful}^{\theta^*} \,\&\, \pi_i^* \in Path_{fin}^{\theta^*}\}=Pr_{fin}^{\theta^*}(\pi_i^*)$.

5 基于概率知识区域图的限界模型检测

本节我们研究如何将 PTCTLK 的模型检测问题归约为区域图上无实时约束的某种时态逻辑的检测问题,然后为缓解状态空间爆炸研究限界模型检测算法.

5.1 时态逻辑的转换

定义 21(PBTLK 的语法). PBTLK 的语法定义如下:

$\phi ::= \text{true} \mid a \mid \phi\wedge\phi \mid \neg\phi \mid z.\phi \mid [\phi\exists U\phi]_{\triangleright\eta} \mid [\phi\forall U\phi]_{\triangleright\eta} \mid [\phi\exists R\phi]_{\triangleright\eta} \mid [\phi\forall R\phi]_{\triangleright\eta} \mid [\exists G\phi]_{\triangleright\eta} \mid [\forall G\phi]_{\triangleright\eta} \mid [K_i\phi]_{\triangleright\eta} \mid [E_\Gamma\phi]_{\triangleright\eta}$,

其中,$a\in Ap$ 是原子命题,$z\in\gamma$,$\triangleright\in\{>,\geqslant\}$,$\Gamma\subseteq Ag$.

将 PTCTLK 公式转换为 PBTLK 公式的主要思路是:将时钟约束ζ转换为原子命题 a_ζ,其余的公式保持不变.

定义 22(PBTLK 的满足性关系). 令 $Region(P,\phi)=(S^*,\bar{s}^*,Act^*,Step^*,Label^*,PK_1^*,...,PK_n^*)$,ϕ是 PBTLK 公式,Θ^*是调度的集合,满足性关系$|=_{\Theta^*}$递归定义如下:

- $s^* |=_{\Theta^*} true$ 对所有的 s^*都成立.
- $s^* |=_{\Theta^*} a$ 当且仅当 $a\in Label^*(s^*)$.
- $s^* |=_{\Theta^*} \neg\phi$ 当且仅当 $s^* |\neq_{\Theta^*} \phi$.
- $s^* |=_{\Theta^*} \phi_1\wedge\phi_2$ 当且仅当 $s^* |=_{\Theta^*} \phi_1, s^* |=_{\Theta^*} \phi_2$.
- $s^* |=_{\Theta^*} z.\phi$ 当且仅当 $(l,\alpha[z:=0]) |=_{\Theta^*} \phi$ (设 $s^*=(l,\alpha)$).
- $s^* |=_{\Theta^*} [\phi_1\exists U\phi_2]_{\triangleright\eta}$ 当且仅当存在调度$\theta^*\in\Theta^*$,使得 $Pr^{\theta^*}(\{\pi^* \mid \pi^* \in Path_{ful}^{\theta^*}(s^*) \,\&\, \pi^* |=_{\Theta^*} \phi_1 U\phi_2\}) \triangleright \eta$.
- $s^* |=_{\Theta^*} [\phi_1\forall U\phi_2]_{\triangleright\eta}$ 当且仅当对任意的调度$\theta^*\in\Theta^*$, $Pr^{\theta^*}(\{\pi^* \mid \pi^* \in Path_{ful}^{\theta^*}(s^*) \,\&\, \pi^* |=_{\Theta^*} \phi_1 U\phi_2\}) \triangleright \eta$.
- $\pi^* |=_{\Theta^*} \phi_1 U\phi_2$ 当且仅当存在 $j\in\mathbb{N}$,使得 $\pi^*(j) |=_{\Theta^*} \phi_2$,且对所有的 $i<j, \pi^*(i) |=_{\Theta^*} \phi_1$.
- $s^* |=_{\Theta^*} [\exists G\phi]_{\triangleright\eta}$ 当且仅当存在调度$\theta^*\in\Theta^*$,使得 $Pr^{\theta^*}(\{\pi^* \mid \pi^* \in Path_{ful}^{\theta^*}(s^*) \,\&\, \pi^* |=_{\Theta^*} G\phi\}) \triangleright \eta$.
- $s^* |=_{\Theta^*} [\forall G\phi]_{\triangleright\eta}$ 当且仅当对任意的调度$\theta^*\in\Theta^*$, $Pr^{\theta^*}(\{\pi^* \mid \pi^* \in Path_{ful}^{\theta^*}(s^*) \,\&\, \pi^* |=_{\Theta^*} G\phi\}) \triangleright \eta$.
- $\pi^* |=_{\Theta^*} G\phi$ 当且仅当对任意的 $j\in\mathbb{N}, \pi^*(j) |=_{\Theta^*} \phi$.
- $s^* |=_{\Theta^*} [\phi_1\exists R\phi_2]_{\triangleright\eta}$ 当且仅当存在调度$\theta^*\in\Theta^*$,使得 $Pr^{\theta^*}(\{\pi^* \mid \pi^* \in Path_{ful}^{\theta^*}(s^*) \,\&\, \pi^* |=_{\Theta^*} \phi_1 R\phi_2\}) \triangleright \eta$.
- $s^* |=_{\Theta^*} [\phi_1\forall R\phi_2]_{\triangleright\eta}$ 当且仅当对任意的调度$\theta*\in\Theta^*$, $Pr^{\theta^*}(\{\pi^* \mid \pi^* \in Path_{ful}^{\theta^*}(s^*) \,\&\, \pi^* |=_{\Theta^*} \phi_1 R\phi_2\}) \triangleright \eta$.
- $\pi^* |=_{\Theta^*} \phi_1 R\phi_2$ 当且仅当:1) 存在 $j\in\mathbb{N}$,使得 $\pi^*(j) |=_{\Theta^*} \phi_1$,且对所有的 $i\leqslant j, \pi^*(i) |=_{\Theta^*} \phi_2$;或者 2) 对任意的 $j\in\mathbb{N}, \pi^*(j) |=_{\Theta^*} \phi_2$.

- $s^* \models_{\Theta^*} [K_i\phi]_{\rhd\eta}$ 当且仅当 $\sum_{s_1^* \models_{\Theta^*} \phi} PK_i^*(s^*, s_1^*) \rhd \eta$.
- $s^* \models_{\Theta^*} [E_\Gamma\phi]_{\rhd\eta}$ 当且仅当对任意的 $i \in \Gamma$, $\sum_{s_1^* \models_{\Theta^*} \phi} PK_i^*(s^*, s_1^*) \rhd \eta$.

定理 1(模型检测的正确性). 给定概率时间自动机 $P=(L,\bar{l},\chi,Act,inv,enab,prob,\ell)$,对应的概率实时解释系统 $PM=(S,\bar{s},\mathbb{R}\times Act,Step,Label,\sim_1,...,\sim_n,PK_1,...,PK_n)$,$PM$ 上的调度集合 Θ,PTCTLK 公式 ϕ,转换 ϕ 得到的 PBTLK 公式 ψ.

$\bar{s},0 \models_\Theta \phi$ 当且仅当在对应的概率知识区域图 $Region(P,\phi)=(S^*,\bar{s}^*,Act^*,Step^*,Label^*,PK_1^*,...,PK_n^*)$ 中,

$$\bar{s}^* \models_{\Theta^*} \psi.$$

证明:采用对公式 ϕ 的长度进行归纳完成证明.对于原子命题,公式的否定与合取形式结论显然成立.我们选取 U 算子来证明结论成立,其他算子的证明类似,在此不再赘述.

设 $\bar{s},0 \models_\theta [\phi_1 \exists U \phi_2]_{\rhd\eta}$,由定义 13 可知,存在调度 $\theta \in \Theta$,使得 $Pr^\theta(\{\pi \mid \pi \in Path_{ful}^\theta(\bar{s}) \& \pi,0 \models_\Theta \phi_1 U \phi_2\}) \rhd \eta$. 再由定义 13 可知,$\pi,0 \vDash_\Theta \phi_1 U \phi_2$ 当且仅当存在位置 (j,t),使得 $\pi(j)+t, D_\pi(j)+t \vDash_\Theta \phi_2$,对所有的位置 (j',t'),如果 $(j',t')<(j,t)$,则 $\pi(j')+t', D_\pi(j')+t' \vDash_\Theta \phi_1$.由归纳假设可知,$\pi^*(j) \models_{\Theta^*} \phi_2$,对任意的 $j'<j$, $\pi^*(j') \models_{\Theta^*} \phi_1$,即 $\pi^* \models_{\Theta^*} \phi_1 U \phi_2$.

因此,由概率知识区域图的定义可知:

$$Pr^\theta(\{\pi \mid \pi \in Path_{ful}^\theta(\bar{s}) \& \pi,0 \models_\Theta \phi_1 U \phi_2\}) = Pr^{\theta^*}(\{\pi^* \mid \pi^* \in Path_{ful}^{\theta^*}(\bar{s}^*) \& \pi^* \models_{\Theta^*} \phi_1 U \phi_2\}).$$

即 $\bar{s}^* \models_{\Theta^*} [\phi_1 \exists U \phi_2]_{\rhd\eta}$.

设 $\bar{s}^* \models_{\Theta^*} [\phi_1 \exists U \phi_2]_{\rhd\eta}$,由定义 22 可知,存在调度 $\theta^* \in \Theta^*$,使得 $Pr^{\theta^*}(\{\pi^* \mid \pi^* \in Path_{ful}^{\theta^*}(\bar{s}^*) \& \pi^* \models_{\Theta^*} \phi_1 U \phi_2\}) \rhd \eta$.

再由定义 22 可知,$\pi^* \models_{\Theta^*} \phi_1 U \phi_2$ 当且仅当存在 $j \in \mathbb{N}$,使得 $\pi^*(j) \models_{\Theta^*} \phi_2$,且对所有的 $i<j$, $\pi^*(i) \models_{\Theta^*} \phi_1$.由归纳假设和概率知识区域图的定义可知,存在位置 (j,t),使得 $\pi(j)+t, D_\pi(j)+t \vDash_\Theta \phi_2$,对所有的位置 (j',t'),如果 $(j',t')<(j,t)$,则 $\pi(j')+t', D_\pi(j')+t' \vDash_\Theta \phi_1$.因此,

$$Pr^{\theta^*}(\{\pi^* \mid \pi^* \in Path_{ful}^{\theta^*}(\bar{s}^*) \& \pi^* \models_{\Theta^*} \phi_1 U \phi_2\}) = Pr^\theta(\{\pi \mid \pi \in Path_{ful}^\theta(\bar{s}) \& \pi,0 \models_\Theta \phi_1 U \phi_2\}).$$

即 $\bar{s},0 \models_\theta [\phi_1 \exists U \phi_2]_{\rhd\eta}$. □

5.2 PBTLK的限界模型检测

限界模型检测的主要思想是在系统有限的局部空间中寻找属性成立的证据或者反例.对于 PBTLK 中的计算树逻辑部分,我们可以采用 LTL 限界模型检测中的技术来定义其限界语义.对于概率算子部分,限界语义必须保证属性在有限局部空间中成立,在整个运行空间中也一定成立.对于算子 $[]_{\geqslant p}$,如果在有限局部空间中属性成立的概率不小于实数 p,自然地,在整个运行空间上属性成立的概率也不小于 p.而对于 $[]_{\leqslant p}$ 这类算子,如果在有限局部空间中属性成立的概率不大于实数 p,则并不能保证在整个运行空间上属性成立的概率也不大于 p.为了保证 $P_{\leqslant p}$ 算子限界语义定义的正确性,本节我们探讨如何将 PBTLK 公式转换为等价的且概率约束为 $[]_{\geqslant p}$ 或者 $[]_{>p}$ 形式的 PBTLK 公式.

定义 23(PBTLK 公式的等价). 称 PBTLK 状态公式 ϕ,φ 是等价的,记为 $\phi \equiv \varphi$,当且仅当对任意的 $Region(P,\phi)=(S^*,\bar{s}^*,Act^*,Step^*,Label^*,PK_1^*,...,PK_n^*)$、任意的 $s^* \in S^*$,$s^* \vDash \phi$ 当且仅当 $s^* \vDash \varphi$.

不难验证下面的等价关系:

- $[\exists G\phi]_{\leqslant p} \equiv [true \exists U(\neg\phi)]_{\geqslant 1-p}$; $[\forall G\phi]_{\leqslant p} \equiv [true \forall U(\neg\phi)]_{\geqslant 1-p}$;
- $[\exists G\phi]_{<p} \equiv [true \exists U(\neg\phi)]_{>1-p}$; $[\forall G\phi]_{<p} \equiv [true \forall U(\neg\phi)]_{>1-p}$;
- $[\phi \exists U\varphi]_{\leqslant p} \equiv [true \exists R(\neg\varphi)]_{\geqslant 1-p}$; $[\phi \forall U\varphi]_{\leqslant p} \equiv [(\neg\phi) \forall R(\neg\varphi)]_{\geqslant 1-p}$;
- $[\phi \exists U\varphi]_{<p} \equiv [true \exists R(\neg\varphi)]_{>1-p}$; $[\phi \forall U\varphi]_{<p} \equiv [(\neg\phi) \forall R(\neg\varphi)]_{>1-p}$;
- $[\phi \exists R\varphi]_{\leqslant p} \equiv [(\neg\phi) \exists U(\neg\varphi)]_{\geqslant 1-p}$; $[\phi \forall R\varphi]_{\leqslant p} \equiv [(\neg\phi) \forall U(\neg\varphi)]_{\geqslant 1-p}$;

- $[\phi \exists R\varphi]_{<p} \equiv [(\neg\phi)\exists U(\neg\varphi)]_{\geqslant 1-p}$; $[\phi \exists R\varphi]_{\leqslant p} \equiv [(\neg\phi)\exists U(\neg\varphi)]_{>1-p}$;
- $[K_i\phi]_{\leqslant p} \equiv [K_i\neg\phi]_{\geqslant 1-p}$; $[K_i\phi]_{<p} \equiv [K_i\neg\phi]_{>1-p}$;
- $[E_\Gamma\phi]_{\leqslant p} \equiv [E_\Gamma\neg\phi]_{\geqslant 1-p}$; $[E_\Gamma\phi]_{<p} \equiv [E_\Gamma\neg\phi]_{>1-p}$.

上面的等价关系说明可将$\leqslant(<)p$ 的概率约束转换为$\geqslant(>)p$ 的约束.下面的等价关系说明,可将否定算子直接作用于原子命题上,且不会降低 PBTLK 的表达力:

- $\neg[\varphi]_{\leqslant p} \equiv [\varphi]_{>p}$; $\neg[\varphi]_{<p} \equiv [\varphi]_{\geqslant p}$;
- $\neg[\varphi]_{\geqslant p} \equiv [\varphi]_{<p}$; $\neg[\varphi]_{>p} \equiv [\varphi]_{\leqslant p}$;
- $\neg(\phi_1\wedge\phi_2) \equiv \neg\phi_1\vee\neg\phi_2$; $\neg(\phi_1\vee\phi_2) \equiv \neg\phi_1\wedge\neg\phi_2$.

上述两类等价关系表明,我们只需在 PBTLK 的某个子集上讨论其限界模型检测问题.该子集与 PBTLK 具有相同的表达力,且概率约束只能为$\geqslant p$ 或者$>p$;否定算子只能作用于原子命题,我们将该子集记为 PBTLK$^{\geqslant}$.

PBTLK$^{\geqslant}$的语法定义如下:

$$\phi ::= true \mid a \mid \neg a \mid \phi\wedge\phi \mid z.\phi \mid [\phi \exists U\phi]_{\triangleright\eta} \mid [\phi\forall U\phi]_{\triangleright\eta} \mid [\phi\exists R\phi]_{\triangleright\eta} \mid [\phi\forall R\phi]_{\triangleright\eta} \mid [\exists G\phi]_{\triangleright\eta} \mid [\forall G\phi]_{\triangleright\eta} \mid [K_i\phi]_{\triangleright\eta} \mid [E_\Gamma\phi]_{\triangleright\eta},$$

其中,$a\in Ap$ 是原子命题,$z\in\gamma$,$\triangleright\in\{>,\geqslant\}$,$\Gamma\subseteq Ag$.

给定概率知识区域图 $Region(P,\phi)=(S^*,\bar{s}^*,Act^*,Step^*,Label^*,PK_1^*,...,PK_n^*)$,定义 $Reach(\bar{s}^*,k)$ 为从初始状态 $\bar{s}^*$ 出发、k 步内可达的状态集.形式化定义为 $Reach(\bar{s}^*,k)=\{s^* \mid \exists\pi^*\in Path_{ful}^{\theta^*}(\bar{s}^*)\exists i\leq k(\pi^*(i)=s^*)\}$.

定义 24(PBTLK$^{\geqslant}$的满足性关系). 令 $Region(P,\phi)=(S^*,\bar{s}^*,Act^*,Step^*,Label^*,PK_1^*,...,PK_n^*)$,$\phi$是 PBTLK$^{\geqslant}$公式,$\Theta^*$是调度的集合,$k\in N$,满足性关系 $\models_{\Theta^*,k}$ 递归定义如下:

- $s^*\models_{\Theta^*,k} true$ 对所有的 s^*都成立.
- $s^*\models_{\Theta^*,k} a$ 当且仅当 $a\in Label^*(s^*)$.
- $s^*\models_{\Theta^*,k} \neg a$ 当且仅当 $a\notin Label^*(s^*)$.
- $s^*\models_{\Theta^*,k} \phi_1\wedge\phi_2$ 当且仅当 $s^*\models_{\Theta^*,k}\phi_1$ 且 $s^*\models_{\Theta^*,k}\phi_2$.
- $s^*\models_{\Theta^*} z.\phi$ 当且仅当 $l,\alpha[z:=0]\models_{\Theta^*,k}\phi$ (设 $s^*=(l,\alpha)$).
- $s^*\models_{\Theta^*,k}[\phi_1\exists U\phi_2]_{\triangleright\eta}$ 当且仅当存在调度$\theta^*\in\Theta^*$,使得 $Pr^{\theta^*}(\{\pi^* \mid \pi^*\in Path_{ful}^{\theta^*}(s^*)\ \&\ \pi^*\models_{\Theta^*,k}\phi_1U\phi_2\})\triangleright\eta$.
- $s^*\models_{\Theta^*,k}[\phi_1\forall U\phi_2]_{\triangleright\eta}$ 当且仅当对任意的调度$\theta^*\in\Theta^*$, $Pr^{\theta^*}(\{\pi^* \mid \pi^*\in Path_{ful}^{\theta^*}(s^*)\ \&\ \pi^*\models_{\Theta^*,k}\phi_1U\phi_2\})\triangleright\eta$.
- $\pi^*\models_{\Theta^*,k}\phi_1U\phi_2$ 当且仅当存在$j\leqslant k$,使得 $\pi^*(j)\models_{\Theta^*,k}\phi_2$, 对所有的 $i<j$, $\pi^*(i)\models_{\Theta^*,k}\phi_1$.
- $s^*\models_{\Theta^*,k}[\exists G\phi]_{\triangleright\eta}$ 当且仅当存在调度$\theta^*\in\Theta^*$,使得 $Pr^{\theta^*}(\{\pi^* \mid \pi^*\in Path_{ful}^{\theta^*}(s^*)\ \&\ \pi^*\models_{\Theta^*,k}G\phi\})\triangleright\eta$.
- $s^*\models_{\Theta^*,k}[\forall G\phi]_{\triangleright\eta}$ 当且仅当对任意的调度$\theta^*\in\Theta^*$, $Pr^{\theta^*}(\{\pi^* \mid \pi^*\in Path_{ful}^{\theta^*}(s^*)\ \&\ \pi^*\models_{\Theta^*,k}G\phi\})\triangleright\eta$.
- $\pi^*\models_{\Theta^*,k}G\phi$ 当且仅当对任意的$j\leqslant k$, $\pi^*(j)\models_{\Theta^*,k}\phi$, 且存在$i\leqslant k$,使得
$$\pi^*=\pi^*(0)\ldots\pi^*(i-1)(\pi^*(i)\ldots\pi^*(k))^\omega.$$
- $s^*\models_{\Theta^*,k}[\phi_1\exists R\phi_2]_{\triangleright\eta}$ 当且仅当存在调度$\theta^*\in\Theta^*$,使得 $Pr^{\theta^*}(\{\pi^* \mid \pi^*\in Path_{ful}^{\theta^*}(s^*)\ \&\ \pi^*\models_{\Theta^*,k}\phi_1R\phi_2\})\triangleright\eta$.
- $s^*\models_{\Theta^*,k}[\phi_1\forall R\phi_2]_{\triangleright\eta}$ 当且仅当对任意的调度$\theta^*\in\Theta^*$, $Pr^{\theta^*}(\{\pi^* \mid \pi^*\in Path_{ful}^{\theta^*}(s^*)\ \&\ \pi^*\models_{\Theta^*,k}\phi_1R\phi_2\})\triangleright\eta$.
- $\pi^*\models_{\Theta^*,k}\phi_1R\phi_2$ 当且仅当:1) 存在$j\leqslant k$,使得 $\pi^*(j)\models_{\Theta^*,k}\phi_1$,对所有的 $i\leqslant j$, $\pi^*(i)\models_{\Theta^*,k}\phi_2$;或者 2) 对任意的 $j\leqslant k$, $\pi^*(j)\models_{\Theta^*,k}\phi_2$,且存在 $i\leqslant k$,使得$\pi^*=\pi^*(0)\ldots\pi^*(i-1)(\pi^*(i)\ldots\pi^*(k))^\omega$.
- $s^*\models_{\Theta^*,k}[K_i\phi]_{\triangleright\eta}$ 当且仅当 $\sum_{s_1^*\models\phi\wedge s_1^*\in Reach(\bar{s}^*,k)} PK_i^*(s^*,s_1^*)\triangleright\eta$.
- $s^*\models_{\Theta^*,k}[E_\Gamma\phi]_{\triangleright\eta}$ 当且仅当对所有的 $i\in\Gamma$, $\sum_{s_1^*\models\phi\wedge s_1^*\in Reach(\bar{s}^*,k)} PK_i^*(s^*,s_1^*)\triangleright\eta$.

定理 2. 令 $Region(P,\phi)=(S^*,\bar{s}^*,Act^*,Step^*,Label^*,PK_1^*,...,PK_n^*)$,$\phi$是 PBTLK$^{\geqslant}$公式,$\Theta^*$是调度的集合,$k\in\mathbb{N}$,如果 $s^*\models_{\Theta^*,k}\phi$,则 $s^*\models_{\Theta^*}\phi$.

证明:采用对公式ϕ的长度进行归纳完成证明.对于原子命题及其否定形式,公式的合取形式结论显然成立.我们选取 U 与 K_i 算子来证明结论成立,其他算子的证明类似,在此不再赘述.

Case 1. U 算子

设 $s^*\models_{\Theta^*,k}[\phi_1\exists U\phi_2]_{\triangleright\eta}$.由定义 24 可知,存在调度 $\theta^*\in\Theta^*$,使得 $Pr^{\theta^*}(\{\pi^*\mid\pi^*\in Path_{ful}^{\theta^*}(s^*)\ \&\ \pi^*\models_{\Theta^*,k}\phi_1U\phi_2\})\triangleright\eta$.由归纳假设可知,如果 $\pi^*\models_{\Theta^*,k}\phi_1U\phi_2$,则 $\pi^*\models_{\Theta^*}\phi_1U\phi_2$,因此,

$$Pr^{\theta^*}(\{\pi^*\mid\pi^*\in Path_{ful}^{\theta^*}(s^*)\ \&\ \pi^*\models_{\Theta^*,k}\phi_1U\phi_2\})\leqslant Pr^{\theta^*}(\{\pi^*\mid\pi^*\in Path_{ful}^{\theta^*}(s^*)\ \&\ \pi^*\models_{\Theta^*}\phi_1U\phi_2\}).$$

从而 $Pr^{\theta^*}(\{\pi^*\mid\pi^*\in Path_{ful}^{\theta^*}(s^*)\ \&\ \pi^*\models_{\Theta^*}\phi_1U\phi_2\})\triangleright\eta$,即 $s^*\models_{\Theta^*}[\phi_1\exists U\phi_2]_{\triangleright\eta}$.

对于 $s^*\models_{\Theta^*,k}[\phi_1\forall U\phi_2]_{\triangleright\eta}$,情形类似不再赘述.

Case 2. K_i 算子

设 $s^*\models_{\Theta^*,k}[K_i\phi]_{\triangleright\eta}$.由定义 24 可知,$\sum_{s_1^*\models\phi\wedge s_1^*\in Reach(\bar{s}^*,k)}PK_i^*(s^*,s_1^*)\triangleright\eta$.

因为 $\sum_{s_1^*\models\phi\wedge s_1^*\in Reach(\bar{s}^*,k)}PK_i^*(s^*,s_1^*)\leqslant\sum_{s_1^*\models\phi}PK_i^*(s^*,s_1^*)$,所以 $\sum_{s_1^*\models\phi}PK_i^*(s^*,s_1^*)\triangleright\eta$,即 $s^*\models_{\Theta^*}[K_i\phi]_{\triangleright\eta}$. □

5.3 限界模型检测算法

本节我们将探讨如何将初始状态对 PBTLK$^{\geqslant}$公式的满足性判定问题转换为线性方程组的求解问题.

对于公式 $[\phi_1\forall U\phi_2]_{\triangleright\eta}$,$[\forall G\phi]_{\triangleright\eta}$,$[\phi_1\forall R\phi_2]_{\triangleright\eta}$,我们主要通过计算 $p_{s^*,k}^{\min}(\phi)=\min\{Pr^{\theta^*}(\{\pi^*\mid\pi^*\in Path_{ful}^{\theta^*}(s^*)\ \&\ \pi^*\models_{\Theta^*,k}\phi_1U\phi_2\})\mid\theta^*\in\Theta^*\}$ 来完成验证过程;

对于公式 $[\phi_1\exists U\phi_2]_{\triangleright\eta}$,$[\exists G\phi]_{\triangleright\eta}$,$[\phi_1\exists R\phi_2]_{\triangleright\eta}$,我们主要通过计算 $p_{s^*,k}^{\max}(\phi)=\max\{Pr^{\theta^*}(\{\pi^*\mid\pi^*\in Path_{ful}^{\theta^*}(s^*)\ \&\ \pi^*\models_{\Theta^*,k}\phi_1U\phi_2\})\mid\theta^*\in\Theta^*\}$ 来完成验证过程.

对 PBTLK$^{\geqslant}$公式ϕ,假设其所有的子公式已经处理过,即对于ϕ的任意子公式φ,对于 S^*中的每一个状态 s^*,均已经知道 s^*是否满足φ.令 $k\geqslant 0$ 为限界模型检测的界,$S_{\phi,k}^*=\{s^*\in S^*\mid s^*\models_{\Theta^*,k}\phi\}$.调度集$\Theta^*$为所有可能的调度的集合.

对 PBTLK$^{\geqslant}$公式ϕ,引入记号 $y(s^*,\phi,k)\in\{0,1\}$来表示 $s^*\models_{\Theta^*,k}\phi$ 是否成立:$y(s^*,\phi,k)=1$ 表示 $s^*\models_{\Theta^*,k}\phi$;0 表示 $s^*\not\models_{\Theta^*,k}\phi$.$y(s^*,\phi,k)$定义如下:

- ϕ是原子命题:如果$\phi\in Label(s^*)$,则 $y(s^*,\phi,k)=1$;否则,$y(s^*,\phi,k)=0$.
- ϕ是原子命题:如果$\phi\in Label^*(s^*)$,则 $y(s^*,\neg\phi,k)=0$;否则,$y(s^*,\neg\phi,k)=1$.
- $\phi=\phi_1\vee\phi_2$:$y(s^*,\phi,k)=y(s^*,\phi_1,k)\vee y(s^*,\phi_2,k)$.
- $\phi=\phi_1\wedge\phi_2$:$y(s^*,\phi,k)=y(s^*,\phi_1,k)\wedge y(s^*,\phi_2,k)$.
- ϕ为 $[\phi_1\forall U\phi_2]_{\triangleright\eta}$,$[\forall G\phi]_{\triangleright\eta}$,$[\phi_1\forall R\phi_2]_{\triangleright\eta}$ 三者之一时:如果 $p_{s^*,k}^{\min}(\phi)\triangleright\eta$,则 $y(s^*,\phi,k)=1$;否则,$y(s^*,\phi,k)=0$.
- ϕ为 $[\phi_1\exists U\phi_2]_{\triangleright\eta}$,$[\exists G\phi]_{\triangleright\eta}$,$[\phi_1\exists R\phi_2]_{\triangleright\eta}$ 三者之一时:如果 $p_{s^*,k}^{\max}(\phi)\triangleright\eta$,则 $y(s^*,\phi,k)=1$;否则,$y(s^*,\phi,k)=0$.
- ϕ为 $[K_i\varphi]_{\triangleright\eta}$,$[E_\Gamma\varphi]_{\triangleright\eta}$ 两者之一时:如果 $p_{s^*,k}(\phi)\triangleright\eta$,则 $y(s^*,\phi,k)=1$;否则,$y(s^*,\phi,k)=0$.

首先讨论 $p_{s^*,k}^{\min}(\phi)$ 和 $p_{s^*,k}^{\max}(\phi)$ 的计算.对于 $p_{s^*,k}^{\min}(\phi)$ 和 $p_{s^*,k}^{\max}(\phi)$,不同的时态算子对应着不同的转换方法,我们分别讨论.首先讨论 $p_{s^*,k}^{\min}(\phi)$ 的计算.

Case 1. ϕ为原子命题

如果$\phi\in Label^*(s^*)$,则 $p_{s^*,k}^{\min}(\phi)=1$;否则,$p_{s^*,k}^{\min}(\phi)=0$.

Case 2. $\phi=G\phi_1$

当 k=0,$y(s^*,\phi_1,0)$=0 时, $p^{\min}_{s^*,0}(\phi)=0$.

当 k=0,$y(s^*,\phi_1,0)$=1 时,如果存在$\mu^*\in Step^*(s^*)$,使得$\mu^*(s^*)$<1,则 $p^{\min}_{s^*,0}(\phi)=0$, 否则 $p^{\min}_{s^*,0}(\phi)=1$.

当 $k\geqslant 1$ 时,

$$p^{\min}_{s^*,k}(\phi)=\min_{\mu_0^*\in Step^*(s_0^*),\ldots,\mu_{k+1}^*\in Steps(s_k^*)}\sum_{i=0}^{k}\sum_{s_0^*,\ldots,s_k^*\in S}y(s_0^*,\phi_1,k)\cdot y(s_1^*,\phi_1,k)\cdot\mu_0^*(s_1^*)\cdot\ldots\cdot y(s_i^*,\phi_1,k)\cdot\mu_i^*(s_i^*)\cdot y(s_{i+1}^*,\phi_1,k)\cdot$$
$$\lfloor\mu_{i+1}^*(s_{i+1}^*)\rfloor\cdot\ldots\cdot y(s_k^*,\phi_1,k)\cdot\lfloor\mu_k^*(s_k^*)\rfloor\cdot\lfloor p^{l_{k+1},\alpha_{k+1}}(s_i^*)\rfloor.$$

这里,记号$\lfloor\mu_j^*(s_{j+1}^*)\rfloor$表示对 $\mu_j^*(s_{j+1}^*)$ 取整(i+1≤j≤k+1).

Case 3. $\phi=\phi_1U\phi_2$

当 k=0 时,如果 $y(s^*,\phi_2,0)$=1,则 $p^{\min}_{s^*,0}(\phi)=1$;否则 $p^{\min}_{s^*,0}(\phi)=0$,即 $p^{\min}_{s^*,0}(\phi)=y(s^*,\phi_2,0)$.

当 $k\geqslant 1$ 时, $p^{\min}_{s^*,k}(\phi)=y(s^*,\phi_2,k)+(1-y(s^*,\phi_2,k))\cdot y(s^*,\phi_1,k)\cdot\min_{\mu^*\in Step^*(s^*)}\left\{\sum_{s_1^*\in S}\mu^*(s_1^*)p^{\min}_{s_1^*,k-1}(\phi)\right\}$.

Case 4. $\phi=\phi_1R\phi_2$

当 k=0 时:

- 如果 $y(s^*,\phi_1,0)$=$y(s^*,\phi_2,0)$=1,则 $p^{\min}_{s^*,0}(\phi)=1$.
- 如果 $y(s^*,\phi_1,0)$=0,$y(s^*,\phi_2,0)$=1,则当存在$\mu^*\in Step^*(s^*)$,使得$\mu^*(s^*)$<1 时, $p^{\min}_{s^*,0}(\phi)=0$;否则, $p^{\min}_{s^*,0}(\phi)=1$.
- 如果 $y(s^*,\phi_2,0)$=0,则 $p^{\min}_{s^*,0}(\phi)=0$.

当 $k\geqslant 1$ 时,因为$\phi=\varphi R\gamma\equiv G\gamma\vee(\gamma U(\gamma\wedge\varphi))$,故分成两部分:

$$p^{\min}_{s^*,k}(\phi)=\min_{\mu_0^*\in Step^*(s_0^*),\ldots,\mu_{k+1}^*\in Steps(s_k^*)}\sum_{i=0}^{k}\sum_{s_0^*,\ldots,s_k^*\in S}y(s_0^*,\phi_1,k)\cdot y(s_1^*,\phi_1,k)\cdot\mu_0^*(s_1^*)\cdot\ldots\cdot y(s_i^*,\phi_1,k)\cdot\mu_i^*(s_i^*)\cdot y(s_{i+1}^*,\phi_1,k)\cdot$$
$$\lfloor\mu_{i+1}^*(s_{i+1}^*)\rfloor\cdot\ldots\cdot y(s_k^*,\phi_1,k)\cdot\lfloor\mu_k^*(s_k^*)\rfloor\cdot\lfloor p^{l_{k+1},\alpha_{k+1}}(s_i^*)\rfloor+p^{\min}_{s^*,k}(\gamma U(\gamma\wedge\varphi)).$$

这里,加入因子 $(1-y(s_j^*,\phi_1,k))(0\leqslant j\leqslant k)$ 的主要目的是避免重复计算$\{\pi^*|\pi^*\vDash_k G\phi_2\wedge\pi\vDash_k\phi_2U(\phi_1\wedge\phi_2)\}$的概率度量.

现在讨论 $p^{\max}_{s^*,k}(\phi)$ 的计算.

Case 1. ϕ为原子命题

如果$\phi\in Label^*(s^*)$,则 $p^{\max}_{s^*,k}(\phi)=1$;否则, $p^{\max}_{s^*,k}(\phi)=0$.

Case 2. $\phi=G\phi_1$

当 k=0,$y(s^*,\phi_1,0)$=0 时, $p^{\max}_{s^*,0}(\phi)=0$.

当 k=0,$y(s^*,\phi_1,0)$=1 时,如果存在$\mu^*\in Step^*(s^*)$,使得$\mu^*(s^*)$=1,则 $p^{\max}_{s^*,0}(\phi)=1$;否则, $p^{\max}_{s^*,0}(\phi)=0$.

当 $k\geqslant 1$ 时,

$$p^{\max}_{s^*,k}(\phi)=\max_{\mu_0^*\in Step^*(s_0^*),\ldots,\mu_{k+1}^*\in Steps(s_k^*)}\sum_{i=0}^{k}\sum_{s_0^*,\ldots,s_k^*\in S}y(s_0^*,\phi_1,k)\cdot y(s_1^*,\phi_1,k)\cdot\mu_0^*(s_1^*)\cdot\ldots\cdot y(s_i^*,\phi_1,k)\cdot\mu_i^*(s_i^*)\cdot y(s_{i+1}^*,\phi_1,k)\cdot$$
$$\lfloor\mu_{i+1}^*(s_{i+1}^*)\rfloor\cdot\ldots\cdot y(s_k^*,\phi_1,k)\cdot\lfloor\mu_k^*(s_k^*)\rfloor\cdot\lfloor p^{l_{k+1},\alpha_{k+1}}(s_i^*)\rfloor.$$

Case 3. $\phi=\phi_1U\phi_2$

当 k=0 时,如果 $y(s^*,\phi_2,0)$=1,则 $p^{\max}_{s^*,0}(\phi)=1$;否则, $p^{\max}_{s^*,0}(\phi)=0$,即 $p^{\min}_{s^*,0}(\phi)=y(s^*,\phi_2,0)$.

当 $k\geqslant 1$ 时, $p^{\max}_{s^*,k}(\phi)=y(s^*,\phi_2,k)+(1-y(s^*,\phi_2,k))\cdot y(s^*,\phi_1,k)\cdot\max_{\mu^*\in Step^*(s^*)}\left\{\sum_{s_1^*\in S}\mu^*(s_1^*)p^{\max}_{s_1^*,k-1}(\phi)\right\}$.

Case 4. $\phi=\phi_1R\phi_2$

当 k=0 时,

- 如果 $y(s^*,\phi_1,0)$=$y(s^*,\phi_2,0)$=1,则 $p^{\max}_{s^*,0}(\phi)=1$.
- 如果 $y(s^*,\phi_1,0)$=0,$y(s^*,\phi_2,0)$=1,则当存在$\mu^*\in Step^*(s^*)$,使得$\mu^*(s^*)$=1 时, $p^{\max}_{s^*,0}(\phi)=0$;否则, $p^{\max}_{s^*,0}(\phi)=1$.
- 如果 $y(s^*,\phi_2,0)$=0,则 $p^{\max}_{s^*,0}(\phi)=0$.

当 $k\geqslant 1$ 时,因为ϕ=$\phi_1 R\phi_2\equiv G\phi_2\vee(\phi_2 U(\phi_2\wedge\phi_1))$,故分成两部分:

$$p^{\max}_{s^*,k}(\phi)=\max_{\mu_0^*\in Step^*(s_0^*),\ldots,\mu_{k+1}^*\in Steps(s_k^*)}\sum_{i=0}^{k}\sum_{s_0^*,\ldots,s_k^*\in S} y(s_0^*,\phi_1,k)\cdot y(s_1^*,\phi_1,k)\cdot\mu_0^*(s_1^*)\cdot\ldots\cdot y(s_i^*,\phi_1,k)\cdot\mu_i^*(s_i^*)\cdot y(s_{i+1}^*,\phi_1,k)\cdot$$
$$\left\lfloor\mu_{i+1}^*(s_{i+1}^*)\right\rfloor\cdot\ldots\cdot y(s_k^*,\phi_1,k)\cdot\left\lfloor\mu_k^*(s_k^*)\right\rfloor\cdot\left\lfloor p^{l_{k+1},\alpha_{k+1}}(s_i^*)\right\rfloor+p^{\max}_{s^*,k}(\gamma U(\gamma\wedge\varphi)).$$

这里,加入因子 $(1-y(s_j^*,\phi_1,k))(0\leqslant j\leqslant k)$ 的主要目的是避免重复计算$\{\pi^*|\pi^*\models_k G\phi_2\wedge\pi\models_k\phi_2 U(\phi_1\wedge\phi_2)\}$的概率度量.

现在讨论知识算子对应的概率度量的计算.

Case 1. $\phi=[K_i\varphi]_{\rhd\eta}$

当 k=0 时, $p_{s^*,0}(\phi)=y(s^*,\varphi,0)\cdot PK_i^*(s^*)(s^*)$.

当 $k\geqslant 1$ 时, $p_{s^*,k}(\phi)=\sum_{s_1^*\in Reach(s^*,k)} y(s_1^*,\varphi,k)\cdot PK_i^*(s^*)(s_1^*)$.

Case 2. $\phi=[E_\Gamma\varphi]_{\rhd\eta}$

当 k=0 时, $p_{s^*,0}(\phi)=\min\{y(s^*,\varphi,0)\cdot PK_i^*(s^*)(s^*)\,|\,i\in\Gamma\}$.

当 $k\geqslant 1$ 时, $p_{s^*,k}(\phi)=\min\left\{\sum_{s_1^*\in Reach(s^*,k)} y(s_1^*,\varphi,k)\cdot PK_i^*(s^*)(s_1^*)\,|\,i\in\Gamma\right\}$.

现在分析变元数与模型、界、公式大小之间的依赖关系.

定义 25(*l* 步可达). 对于状态 s^*:1) 如果 $s_1^*=s^*$,则称 s_1^* 是从 s^*出发 0 步可达的;2) 如果 s_{l-1}^* 是从 s^*出发 l–1 步可达的,且存在 $\mu^*\in Step^*(s_{l-1}^*)$,使得 $\mu^*(s_l^*)>0$,则称 s_l^* 是从 s^*出发 l 步可达的.

对 PBTLK$^{\geqslant}$公式ϕ,令$|\phi|$表示ϕ中出现的符号的数目.令 $Region(P,\phi)=(S^*,\bar{s}^*,Act^*,Step^*,Label^*,PK_1^*,\ldots,PK_n^*)$ 为概率知识区域图,N_i 表示从初始状态出发 i 步可达状态的数目,k 为界,ϕ是需要验证的公式,V 表示依据模型检测算法得到的方程组中变元的数目.在每个状态下,ϕ的每一个子公式与每一个不大于 k 的界的组合都可能与一个变元对应.另外,对每一个ϕ的子公式φ,引入了变元 $y(s^*,\varphi,k)$.因此,V 与 $k,N_0,\ldots,N_k,|\phi|$之间的关系为

$$V\leqslant(N_0+\ldots+N_k)\times|\phi|\times k\times 2.$$

6 线性方程组的求解

线性方程组的解法一般分为两类:一类是直接法,就是在没有舍入误差的情况下,通过有限步四则运算可以求得方程组准确解的方法.目前,较实用的直接法都是古老的高斯消去法的变形;另一类是迭代法,就是先给一个解的初始近似值,然后按一定的法则逐步求出解的各个更准确的近似值的方法,如雅可比迭代法、高斯-赛德尔迭代法以及逐次超松弛法和梯度法.

对于中等规模的 n 阶(n<100)线性方程组,由于直接法的准确性和可靠性,它们是经常被选用的方法.对于较高阶的方程组,由于直接法的计算代价较高,使得迭代法更具竞争力.而对于概率时间自动机,限界模型检测方法得出的方程组是上三角方程组,因此我们可以避免使用高斯消去法将一般的方程组变成上三角方程组.上三角方程组的求解非常简单,计算代价小.鉴于此,尽管方程组的阶会非常大,我们仍然选择直接法来求解方程.下面通过对每一个 PBTLK$^{\geqslant}$公式引入语法树的概念来分析变元求解的先后次序.

定义 26(语法树). PBTLK$^{\geqslant}$公式ϕ的语法树是一棵树,其中,内部结点标记为算子$\neg,\wedge,\vee,z,[\forall U]_{\rhd\eta},[\exists U]_{\rhd\eta}$,$[\forall R]_{\rhd\eta},[\exists R]_{\rhd\eta},[\forall G]_{\rhd\eta},[\exists G]_{\rhd\eta},[K_i]_{\rhd\eta},[E_\Gamma]_{\rhd\eta}$,终端结点标记为原子命题.

公式$[K_1(z.[p\forall U(q\wedge(r_{z<5}))]_{>0.99})]_{\geqslant 0.6}$的语法树如图 13 所示.给定语法树上的结点 v,$fml(v)$表示结点上的公式,$v.left$ 和 $v.right$ 表示 v 的左右孩子结点,$v.child$ 表示 v 的唯一的孩子结点.对于公式ϕ,限界模型检测算法按照ϕ的语法树以深度优先的方式进行计算,即给定结点 v,将计算 $fml(v)$的满足性归为计算 v 的孩子节点的满足性.因此,最终将归结为原子命题对应的概率度量的计算.具体的计算次序如下:

1) 计算终端结点对应的变元的解,即原子命题对应的变元的解.
2) 计算终端结点父节点对应的变元的解,顺序依次为 $p^{\min}_{s^*,k}(\phi), p^{\max}_{s^*,k}(\phi), y(s^*,\phi,k)$.
3) 将父结点记为当前结点,如果所有的当前节点都没有父结点则退出;否则,计算当前节点的父结点对应的变元的解,返回步骤 3)继续.

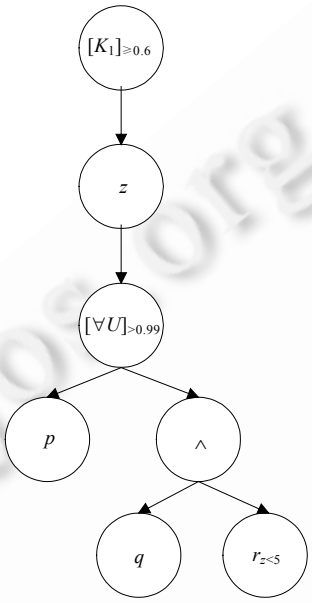

Fig.13 Syntax tree of $[K_1(z.[p\forall U(q\wedge(r_{z<5}))]_{>0.99})]_{\geqslant 0.6}$

图 13 $[K_1(z.[p\forall U(q\wedge(r_{z<5}))]_{>0.99})]_{\geqslant 0.6}$ 的语法树

7 实例研究

7.1 火车穿越控制系统

火车穿越控制系统已经被广泛用来比较实时系统上不同的形式化方法.系统由 Train,Gate,Controller 这 3 个构件组合而成.构件之间平行运行,且通过动作 *approach*,*exit*,*lower*, *raise* 同步协作.我们对标准的穿越控制系统进行一些修改,假设栅栏在执行动作 *lower* 时因为设备劳损的问题会出现无法关闭的可能,具体概率分布如下:成功关闭栅栏的概率是 0.95,无法关闭的概率是 0.05.此外,控制器在发送 *lower* 命令的时候可能会失败,失败的概率是 0.02,成功的概率是 0.98.

- 火车的行为(如图 14 所示)

当火车接近交叉路口时,火车给控制器发送 *approach* 信号,并且必须在 300 秒后给环境发送表示已经进入交叉路口的 *in* 信号.当火车离开交叉路口时,火车发送表示准备离开路口的信号 *out.exit* 信号必须在发出 *approach* 信号后 500 秒内发出,用于与控制器进行同步协作.

- 栅栏的运行(如图 15 所示)

栅栏在接收到 *lower* 信号后,必须在 100 秒内落下栅栏.由于机械故障成功落下栅栏的概率是 0.95,失败的

概率是0.05.栅栏一旦放下,接到 *raise* 信号后就必须在200秒内升起栅栏.由于机械故障栅栏成功升起的概率是0.95,失败的概率是0.05.

- 控制器的运行(如图16所示)

控制器必须在收到 *approach* 信号后100秒时将 *lower* 信号发送给栅栏控制系统.由于设备不可靠,发送成功的概率是0.98,失败的概率是0.02.在接收到 *exit* 信号的100秒内发送信号 *raise*.由于设备不可靠,发送成功的概率是0.95,失败的概率是0.05.

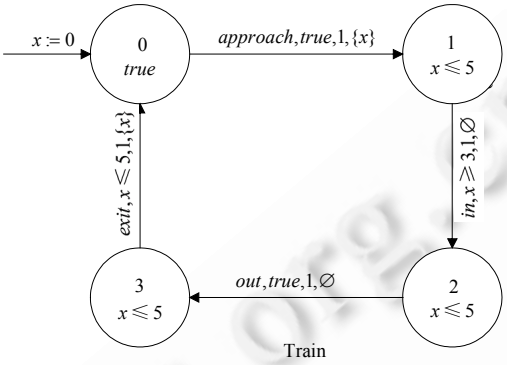

Fig.14 Railroad crossing system: train

图14 火车穿越控制系统:火车

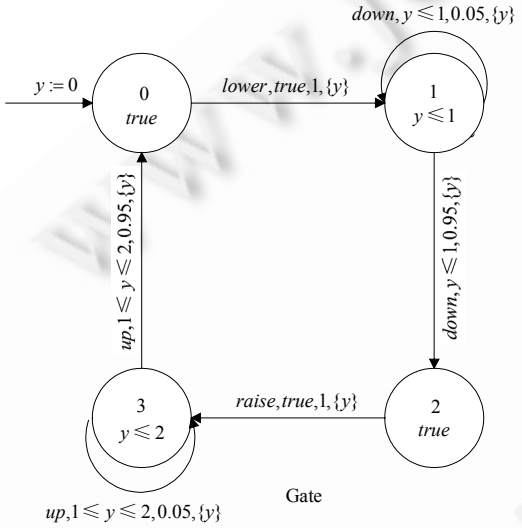

Fig.15 Railroad crossing system: Gate

图15 火车穿越控制系统:栅栏

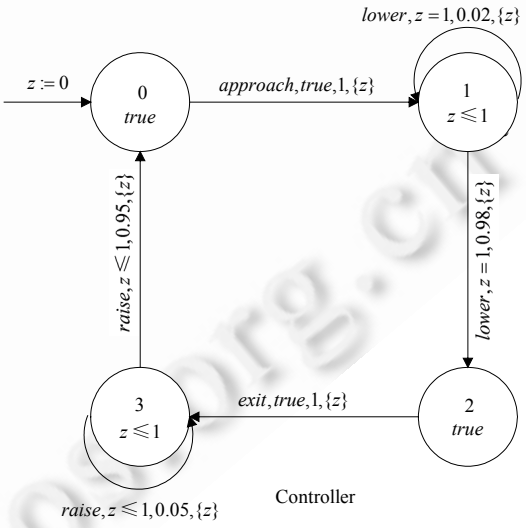

Fig.16 Railroad crossing system: controller

图16 火车穿越控制系统:控制器

7.2 限界模型检测

图17是3个子系统的平行组合.考察属性:火车知道当 *approach* 信号发出去以后,栅栏关闭的概率不低于0.9.利用 PTCTLK 公式该属性描述为$[K_{Train}([true\forall Udown]_{\geqslant 0.9})]_{\geqslant 1}$.我们的目的是验证初始状态 $s_{0,0,0}$ 是否满足$[K_{Train}([true\forall Udown]_{\geqslant 0.9})]_{\geqslant 1}$.现在考察认知关系 K_{Train}.火车随着时间的流逝不停地前进,因此,刻画火车运行时间的变量 x 只有在火车离开之后才会被重置,等待下一辆火车的到来,即对当前火车而言回不到初始状态,从而造成对初始状态((0,0,0),(0,0,0))而言,$K_{Train}(s_{0,0,0})(s_{0,0,0})$=1.令 k=3,考察在3步可达空间中属性是否成立.

$s_{0,0,0}$ 在 3 步可达空间中是否满足$[K_{Train}([true\forall Udown]_{\geqslant 0.9})]_{\geqslant 1}$主要取决于如图 18 所示的概率知识区域图中概率度量 $p^{\max}_{\bar{s}^*,3}(trueUdown)$ 的值.

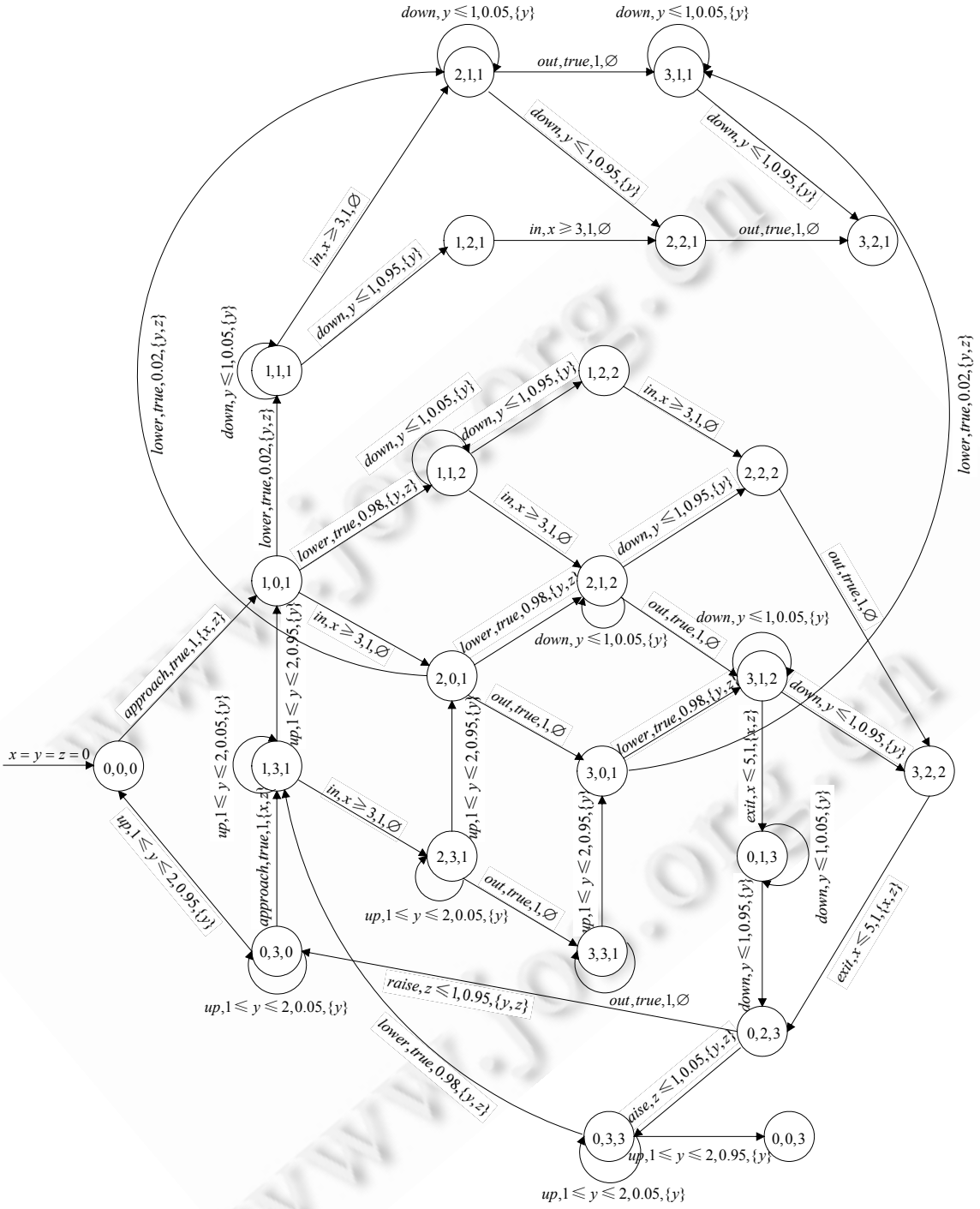

Fig.17 Parallel composition of railroad crossing system

图 17 火车穿越控制系统的平行组合模型

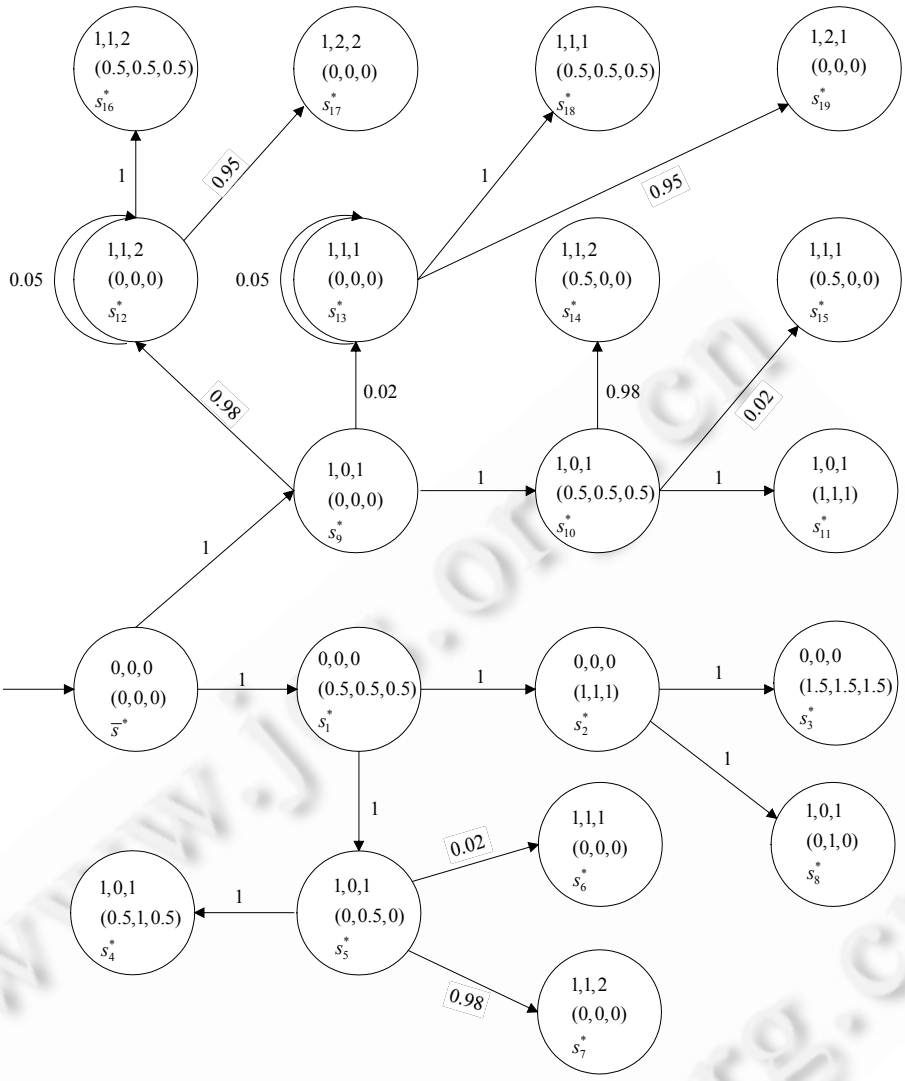

Fig.18 Local probabilistic knowledge region of railroad crossing system

图 18 火车穿越控制系统的部分概率知识区域图

依据限界检测算法,得到的方程组如下:

Step 1. 将步长 3 时 $p^{\max}_{\bar{s}^*,3}(trueUdown)$ 的计算转化为步长为 2 时相应概率度量的计算:

1) $p^{\max}_{\bar{s}^*,3}(trueUdown) = y(\bar{s}^*, down, 3) + (1 - y(\bar{s}^*, down, 3)) \cdot y(\bar{s}^*, true, 3) \cdot$

$$\max\{p^{\max}_{s_1^*,2}(trueUdown), p^{\max}_{s_9^*,2}(trueUdown)\}.$$

Step 2. 将步长 2 时概率度量的计算转化为步长为 1 时概率度量的计算:

2) $p^{\max}_{s_1^*,2}(trueUdown) = y(s_1^*, down, 2) + (1 - y(s_1^*, down, 2)) \cdot y(s_1^*, true, 2) \cdot$

$$\max\{p^{\max}_{s_2^*,1}(trueUdown), p^{\max}_{s_5^*,1}(trueUdown)\}.$$

3) $p^{\max}_{s_9^*,2}(trueUdown) = y(s_9^*, down, 2) + (1 - y(s_9^*, down, 2)) \cdot y(s_9^*, true, 2) \cdot$

$$\max\{p^{\max}_{s_{10}^*,1}(trueUdown), 0.02 \cdot p^{\max}_{s_{13}^*,1}(trueUdown) + 0.98 \cdot p^{\max}_{s_{12}^*,1}(trueUdown)\}.$$

Step 3. 将步长 1 时概率度量的计算转化为步长为 0 时概率度量的计算:

4) $p^{\max}_{s^*_2,1}(trueUdown)=y(s^*_2,down,1)+(1-y(s^*_2,down,1))\cdot y(s^*_2,true,1)\cdot \max\{p^{\max}_{s^*_3,0}(trueUdown),p^{\max}_{s^*_8,0}(trueUdown)\}.$

5) $p^{\max}_{s^*_5,1}(trueUdown)=y(s^*_5,down,1)+(1-y(s^*_5,down,1))\cdot y(s^*_5,true,1)\cdot \max\{0.02\cdot p^{\max}_{s^*_6,0}(trueUdown)+0.98\cdot p^{\max}_{s^*_7,0}(trueUdown)\}.$

6) $p^{\max}_{s^*_{10},1}(trueUdown)=y(s^*_{10},down,1)+(1-y(s^*_{10},down,1))\cdot y(s^*_{10},true,1)\cdot \max\{0.98\cdot p^{\max}_{s^*_{14},0}(trueUdown)+0.02\cdot p^{\max}_{s^*_{15},0}(trueUdown),p^{\max}_{s^*_{11},0}(trueUdown)\}.$

7) $p^{\max}_{s^*_{13},1}(trueUdown)=y(s^*_{13},down,1)+(1-y(s^*_{13},down,1))\cdot y(s^*_{13},true,1)\cdot \max\{0.05\cdot p^{\max}_{s^*_{13},0}(trueUdown)+0.95\cdot p^{\max}_{s^*_{19},0}(trueUdown),p^{\max}_{s^*_{18},0}(trueUdown)\}.$

8) $p^{\max}_{s^*_{12},1}(trueUdown)=y(s^*_{12},down,1)+(1-y(s^*_{12},down,1))\cdot y(s^*_{12},true,1)\cdot \max\{0.05\cdot p^{\max}_{s^*_{12},0}(trueUdown)+0.95\cdot p^{\max}_{s^*_{17},0}(trueUdown),p^{\max}_{s^*_{16},0}(trueUdown)\}.$

Step 4. 将步长 0 时概率度量的计算转化为状态对原子命题的满足性计算:

9) $p^{\max}_{s^*_3,0}(trueUdown)=y(s^*_3,down,0).$

10) $p^{\max}_{s^*_8,0}(trueUdown)=y(s^*_8,down,0).$

11) $p^{\max}_{s^*_{14},0}(trueUdown)=y(s^*_{14},down,0).$

12) $p^{\max}_{s^*_{15},0}(trueUdown)=y(s^*_{15},down,0).$

13) $p^{\max}_{s^*_{11},0}(trueUdown)=y(s^*_{11},down,0).$

14) $p^{\max}_{s^*_{13},0}(trueUdown)=y(s^*_{13},down,0).$

15) $p^{\max}_{s^*_{19},0}(trueUdown)=y(s^*_{19},down,0).$

16) $p^{\max}_{s^*_{18},0}(trueUdown)=y(s^*_{18},down,0).$

17) $p^{\max}_{s^*_{12},0}(trueUdown)=y(s^*_{12},down,0).$

18) $p^{\max}_{s^*_{17},0}(trueUdown)=y(s^*_{17},down,0).$

19) $p^{\max}_{s^*_{16},0}(trueUdown)=y(s^*_{16},down,0).$

Step 5. 计算状态对原子命题的满足性:

20) $y(s^*_3,down,0)=0.$

21) $y(s^*_8,down,0)=0.$

22) $y(s^*_{14},down,0)=0.$

23) $y(s^*_{15},down,0)=0.$

24) $y(s^*_{11},down,0)=0.$

25) $y(s^*_{13},down,0)=0.$

26) $y(s^*_{19},down,0)=1.$

27) $y(s^*_{18},down,0)=0.$

28) $y(s^*_{12},down,0)=0.$

29) $y(s^*_{17},down,0)=1.$

30) $y(s_{16}^*, down, 0) = 0$.

利用直接法解上述方程组可得 $p_{s^*,3}^{\max}(trueUdown) = 0.95$,因此$[K_{Train}([true \forall Udown]_{\geqslant 0.9})]_{\geqslant 1}$ 成立.如果采用无界模型检测技术验证属性$[K_{Train}([true \forall Udown]_{\geqslant 0.9})]_{\geqslant 1}$,则需要遍历整个状态空间,而采用限界检测技术只需遍历可达深度为 3 的状态空间即可.因此,与无界模型检测相比,在属性为真的证据较短的情况下,限界检测完成验证所需空间更小.

8 终止性选择标准

理论上 $s^* |=_{\Theta^*,k} \phi \Rightarrow s^* |=_{\Theta^*,k+1} \phi$,因此随着界的增长,概率度量会逐渐递增.本节我们探索这种递增的规律与限界检测过程终止之间的关系.我们通过考察图 19 的几种曲线来探索这种关系.记号 $Pr(s^*,\phi,\Theta^*,k)$表示步长为 k 时计算得到的概率度量值,ξ为一个预先设置好的非常小的有理数.

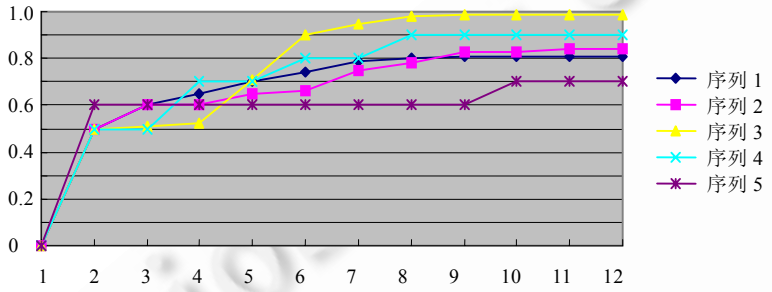

Fig.19 The law of increasing of probability measure

图 19 概率度量增长的规律

判断准则 1. 当 $\Pr(s^*,\phi,\Theta^*,k)-\Pr(s^*,\phi,\Theta^*,k-1) \leqslant \xi$时,计算终止.

准则 1 说明,当两次概率度量计算结果的差控制在ξ内时,计算终止.考察图 19 中的序列 1:0,0.5,0.6,0.65,0.7,0.74,0.79,0.8,0.81,0.81,0.81,0.81,其中,0.81 是精确的概率度量值.令ξ=0.02,算法终止时计算出的概率度量值是 0.8,非常接近 0.81.

判断准则 2. 当 $\Pr(s^*,\phi,\Theta^*,k)-\Pr(s^*,\phi,\Theta^*,k-2) \leqslant \xi$时,计算终止.

准则 2 说明,当间隔 1 次的概率度量计算结果的差控制在ξ内时,计算终止.考察图 19 中的序列 2:0,0.5,0.6,0.6,0.65,0.66,0.75,0.78,0.83,0.83,0.84,0.84,其中,0.84 是精确的概率度量值.令ξ=0.02,如果执行准则 1,则算法终止时概率度量为 0.66,此时距离 0.84 的差距比较大;如果执行标准 2,则算法终止时计算出的概率度量值是 0.84.事实上,准则 2 可以扩展为间隔多次的概率度量计算结果的差.

判断准则 3. 当$|(\Pr(s^*,\phi,\Theta^*,k)-\Pr(s^*,\phi,\Theta^*,k-1))-(\Pr(s^*,\phi,\Theta^*,k-2)-\Pr(s^*,\phi,\Theta^*,k-3))| \leqslant \xi$时,计算终止.

准则 3 首先计算相邻概率度量的差,然后比较间隔 2 次的差之差.考察图 19 中的序列 3:0,0.5,0.51,0.52,0.71,0.9,0.95,0.98,0.99,0.99,0.99,0.99,其中,0.99 是精确的概率度量值.令ξ=0.02,如果执行准则 2,则算法终止时概率度量为 0.52,此时距离精确值 0.99 的差距非常大;如果执行标准 3,则最终的概率度量是 0.98.现在说明为什么准则 3 不设置间隔 1 次的概率度量的差之差作为标准.考察递增的数值序列 x_1,x_2,x_3,准则 2 需要计算 x_3-x_1.此时如果准则 3 的间隔设置为 1 次,则需要计算$|(x_3-x_2)-(x_2-x_1)|$.如果$(x_3-x_2)>(x_2-x_1)$,则 $x_3-x_1-|(x_3-x_2)-(x_2-x_1)|=2x_2-2x_1 \geqslant 0$,否则 $x_3-x_1-|(x_3-x_2)-(x_2-x_1)|=2x_3-2x_2 \geqslant 0$.因此,无论在哪种情况下,准则 3 中设置间隔为 1 次均可以通过准则 2 实现.事实上,准则 3 可以扩展为计算间隔 2 次以上的概率度量计算结果的差之差.

准则 2 是准则 1 的改进,因此如果用准则 2 取代准则 1 则不会降低计算的精度.前面已经说明,对于序列 0,0.5,0.51,0.52,0.71,0.9,0.95,0.98,0.99,0.99,0.99,0.99,准则 3 优于准则 2.但是对于图 19 中的序列 4:0,0.5,0.5,0.7,0.7,0.8,0.8,0.9,0.9,0.9,0.9,0.9,使用准则 2 得到的最终概率度量值是 0.9,而使用准则 3 得到的概率度量值是 0.7.

因此准则 2 优于准则 3,从而两者不可相互替代.事实上,为了进一步提高计算的精度,可以同时使用准则 2 和准则 3.

考察图 19 的序列 5:0,0.6,0.6,0.6,0.6,0.6,0.6,0.6,0.6,0.7,0.7,0.7,上述 3 个规则都将失效,因此数值序列的演化规律与最终的概率度量之间的关系仍需深入研究.概率实时认知逻辑 PTCTLK 的限界模型检测何时终止,依赖于概率实时解释系统的结构、待验证的属性等因素.挖掘这些因素与终止标准的关系从而设置一个合理的终止标准,是一个值得继续研究的问题.

9 结 论

为了解决模型检测概率实时认知逻辑中的状态空间爆炸问题,本文提出了概率实时认知逻辑的限界模型检测技术.围绕限界模型检测的 3 个核心问题,分别提出了有效的解决方案.这些方案不是传统限界模型检测技术的直接推广,而是一种全新的限界模型检测过程,特别是在限界模型检测算法与终止判别标准的设计方面,解决方案的思想完全不同于传统限界检测技术.进一步通过实例,说明了限界模型检测在属性为真的证据较短的情况下需求的空间比无界模型检测技术少.未来的主要工作是对限界模型检测算法进行优化,同时挖掘概率实时解释系统的结构、待验证的属性等因素与终止标准的关系,为设置一个合理的终止标准奠定基础.

References:

[1] Jhala R, McMillan KL. Microarchitecture verification by compositional model checking. Lecture Notes in Computer Science, 2001, 2102:396−410. [doi: 10.1007/3-540-44585-4_40]

[2] McMillan KL. Parameterized verification of the flash cache coherence protocol by compositional model checking. Lecture Notes in Computer Science, 2001,2144:179−195. [doi: 10.1007/3-540-44798-9_17]

[3] Kwiatkowska M, Norman G, Sproston J, Wang FZ. Symbolic model checking for probabilistic timed automata. Information and Computation, 2007,205(7):1027−1077. [doi: 10.1007/978-3-540-30206-3_21]

[4] Ben-Ari M, Pnueli A, Manna Z. The temporal logic of branching time. Acta Information, 1983,20(1):207−226. [doi: 10.1007/BF01257083]

[5] Clarke EM, Grumberg O, Peled D. Model Checking. Cambridge: MIT Press, 1999. 27−34.

[6] Halpern JY. Reasoning About Knowledge: A Survey. In: Gabbay D, Hogger CJ, Robinson JA, eds. Handbook of Logic in Artificial Intelligence and Logic Programming. Vol.4. Oxford: Oxford University Press, 1995. 1−34.

[7] Halpern JY, Vardi MY. Model checking vs. theorem proving: A manifesto. In: Lifschitz V, ed. Proc. of the Artificial Intelligence and Mathematical Theory of Computation. San Diego: Academic Press, 1991. 151−176.

[8] Meyden R, Su KL. Symbolic model checking the knowledge of the dining cryptographers. In: Proc. of the 17th IEEE Computer Security Foundations Workshop. Washington: IEEE Computer Society, 2004. 280−291. [doi: 10.1109/CSFW.2004.1310747]

[9] Luo XY, Su KL, Yang JJ. Bounded model checking for temporal epistemic logic in synchronous multi-agent systems. Journal of Software, 2006,17(12):2485−2498 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/17/2485.htm [doi: 10.1360/jos172485]

[10] Hoek W, Wooldridge M. Model checking knowledge and time. Lecture Notes in Computer Science, 2002,2318:95−111. [doi: 10.1007/3-540-46017-9_9]

[11] Halpern JY, Vardi MY. The complexity of reasoning about knowledge and time. Journal of Computer and System Sciences, 1989,38:195−237. [doi: 10.1145/12130.12161]

[12] Penczek W, Lomuscio A. Verifying epistemic properties of multi-Agent systems via bounded model checking. Fundamenta Informaticae, 2003,55(2):167−185. [doi: 10.1145/860575.860609]

[13] Lomuscio A, Penczek W, Wozna B. Bounded model checking for knowledge and real time. Artificial Intelligence, 2007,171(16-17):1011−1038. [doi: 10.1016/j.artint.2007.05.005]

[14] Ferreira N, Fisher M, Hoek W. Practical reasoning for uncertain agents. Lecture Notes in Computer Science, 2004,3229:82−94. [doi: 10.1007/978-3-540-30227-8_10]

[15] Wan W, Bentahar J, Ben Hamza A. Model checking epistemic and probabilistic properties of multi-agent systems. In: Mehrotra KG, ed. Proc. of the 24th Int'l Conf. on Industrial Engineering and Other Applications of Applied Intelligent Systems Conf. on Modern Approaches in Applied Intelligence. Berlin, Heidelberg: Springer-Verlag, 2011. 68−78. [doi: 10.1007/978-3-642-21827-9_8]

[16] Zhou CH, Sun B, Liu ZF. Abstraction for model checking multi-agent systems. Frontiers of Computer Science in China, 2001,5(1): 14−25. [doi: 10.1007/s11704-010-0358-y]

[17] Cohen M, Dam M, Lomuscio A, Russo F. Abstraction in model checking multi-agent systems. In: Decker, Sichman, Sierra, Castelfranchi, eds. Proc. of the 8th Int'l Conf. on Autonomous Agents and Multiagent Systems. New York: ACM Press, 2009. 945−952. [doi: 10.1145/1558109.1558144]

[18] Lomuscio A, Penczek W, Qu HY. Partial order reductions for model checking temporal epistemic logics over interleaved multi-agent systems. Fundamenta Informaticae, 2010,101(1-2):71−90. [doi: 10.1145/1838206.1838293]

[19] Luo XY, Su KL, Sattar A, Chen QL, Lü GF. Bounded model checking knowledge and branching time in synchronous multi-agent systems. In: Dignum F, Dignum V, Koenig S, Kraus S, Singh MP, Wooldridge M, eds. Proc. of the 4th Int'l Conf. on Autonomous Agents and Multiagent Systems. New York: ACM Press, 2005. 1129−1130. [doi: 10.1145/1082473.1082657]

[20] Biere A, Cimatti A, Clarke EM, Zhu YS. Symbolic model checking without BDDs. Lecture Notes in Computer Science, 1999,1579: 193−207. [doi: 10.1007/3-540-49059-0_14]

[21] Penczek W, Wozna B, Zbrzezny A. Bounded model checking for the universal fragment of CTL. Fundamenta Informaticae, 2002, 51(1-2):135−156.

附中文参考文献:

[9] 骆翔宇,苏开乐,杨晋吉.有界模型检测同步多智体系统的时态认知逻辑.软件学报,2006,17(12):2585−2498. http://www.jos.org.cn/1000-9825/17/2485.htm [doi: 10.1360/jos172485]

周从华(1978－),男,江苏盐城人,博士,副教授,CCF 会员,主要研究领域为形式化方法,模型检测,信息流安全.

王昌达(1971－),男,博士,副教授,CCF 会员,主要研究领域为信息安全技术.

叶萌(1987－),女,硕士生,主要研究领域为模型检测.

刘志锋(1981－),男,博士,讲师,CCF 会员,主要研究领域为形式化方法,模型检测.