软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2012,23(7):1656−1668 [doi: 10.3724/SP.J.1001.2012.04089]

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

概率计算树逻辑的限界模型检测*

周从华+, 刘志锋, 王昌达

(江苏大学 计算机科学与通信工程学院,江苏 镇江 212013)

Bounded Model Checking for Probabilistic Computation Tree Logic

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

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

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

Zhou CH, Liu ZF, Wang CD. Bounded model checking for probabilistic computation tree logic. *Journal of Software*, 2012,23(7):1656−1668 (in Chinese). http://www.jos.org.cn/1000-9825/4089.htm

Abstract: In order to overcome the state explosion problem in model checking the probabilistic computation tree logic, a bounded model checking technique is proposed. First, the bounded semantics of the probabilistic computation tree logic is presented, and then its correctness is proved. Second, by a simple instance the criterion of the traditional termination, based on the length of path, is shown to fail. Based on the termination criterion used in the Newton iteration in numerical computing, a new criterion is given. Third, the bounded model checking procedure of the probabilistic computation tree logic is transformed into linear equations. Finally, three benchmarks are used to present the advantages of the bounded model checking.

Key words: model checking; bounded model checking; probabilistic computation tree logic; Markov chains

摘 要: 为了缓解概率计算树逻辑模型检测中的状态空间爆炸问题,提出了概率计算树逻辑的限界模型检测技术.该技术首先定义概率计算树逻辑的限界语义,并证明其正确性;之后,通过实例说明在传统限界模型检测中,以路径长度作为判断检测过程终止的标准已经失效,基于数值计算中牛顿迭代法的终止准则,设计了新的终止判断标准;然后提出基于线性方程组求解的限界模型检测算法;最后,通过 3 个测试用例说明,概率计算树逻辑限界模型检测方法在反例较短的情况下能够快速完成检测过程,而且比概率计算树逻辑的无界模型检测算法所需求得的状态空间要少.

关键词: 模型检测;限界模型检测;概率计算树逻辑;马尔可夫链
中图法分类号: TP301 **文献标识码**: A

模型检测[1,2]是一种有限状态系统验证技术,因其自动化程度高,逐渐引起了广泛的关注.目前,模型检测已在计算机硬件、通信协议、安全协议的验证等方面获得了较大的成功.传统的模型检测技术利用计算树和线性时态逻辑规约需要验证的属性,这两种时态逻辑可以规范系统行为的绝对正确性,如系统运行不可能失败.然而实际上有很多随机现象,比如不可靠信道上的消息丢失,对这一类现象往往关心某种概率度量,如消息传送失败

* 基金项目: 国家自然科学基金(61003288, 61111130184, 60773049); 江苏省自然科学基金(BK2010192); 教育部博士点基金(20093227110005)

收稿时间: 2010-10-05; 定稿时间: 2011-07-08

的概率不高于 1%等等.对这类属性,计算树和线性时态逻辑是无法刻画的,因此,研究人员在计算树和线性时态逻辑的基础上引入了概率算子,得到了概率计算树逻辑 PCTL[3]等规约形式,并提出了相应的概率模型检测方法[4,5].

概率模型检测一般以有限马尔可夫链作为系统的模型.与传统模型检测一样,状态空间爆炸问题(系统的规模随着并发分量的增加呈指数级增长)[6,7]是概率模型检测实用化的主要瓶颈.为克服该问题,研究人员将传统模型检测中的基于 OBDD 的符号化技术[8−10]、谓词抽象[11]、偏序归约[12]、对称归约[13]、组合推理[14]等状态空间约简技术应用到了概率模型检测上,并获得了很好的效果.限界模型检测[15−20]是近几年出现的一种新的空间约简技术,它成功地缓解了计算树逻辑 CTL[18−20]、线性时态逻辑 LTL[15]、时态认知逻辑 CTLK[17]在模型检测过程中出现的状态空间爆炸问题.因此,将传统限界模型检测技术的思想应用到缓解概率计算树逻辑模型检测中出现的状态空间爆炸问题,必将收到良好的效果,这也是本文主要的研究动机.

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、检测过程终止的判别条件、限界模型检测算法.本文围绕这 3 个问题对概率计算树逻辑 PCTL[3]的限界模型检测进行了系统的研究,具体工作包括 3 个方面:1) 将 PCTL 转换为概率约束仅为$\geqslant p$ 或者$>p$(p 表示事件发生的概率)形式的等价形式,定义了 PCTL 的限界语义,并证明了其正确性;2) 摒弃以路径长度作为终止标准的判别条件,基于数值计算中牛顿迭代法使用的迭代过程终止标准,设计了新的检测过程终止判别条件.即,预先设置一个非常小的有理数ξ,当相连两次限界模型检测得到的概率度量的差控制在ξ内时,检测过程终止;3) 设计了基于线性方程组求解的限界模型检测算法,即将 PCTL 的限界模型检测问题转换为线性方程组的求解问题,从而可以借助于数值计算工具 matlab 完成检测过程.另外,为了提高概率计算的精度,提出了两种 PCTL 限界模型检测过程终止判断标准的修正方案.实验结果表明:1) 随着检测步长的增加,限界模型检测得到的概率度量越来越逼近真实的概率度量;2) PCTL 的限界模型检测是一种前向搜索状态空间的方法,在属性为真的证据比较短的情况下,能够快速地验证属性,而且需求的状态空间少于概率计算树逻辑的无界模型检测算法.

1 相关工作

传统限界模型检测技术有效地缓解了计算树逻辑 CTL、线性时态逻辑 LTL、时态认知逻辑 CTLK 在模型检测过程中出现的状态空间爆炸问题.我们的研究不是对传统限界模型检测技术的简单推广,差别主要体现在 3 个方面:其一、传统限界模型检测以单一路径为分析对象,而我们的方法因为涉及到概率度量,因此将以路径集合为分析对象.概率度量的计算,使得在定义限界语义时必须考虑路径集合的量化问题;其二、传统限界模型检测通常预先给定一个路径的长度,以判断检测过程何时终止,我们通过一个通信协议说明这种方法不再适用于 PCTL 的限界模型检测.为此,我们基于牛顿迭代方法中使用的计算过程终止判断标准,设计了一套完全不同的判断检测过程终止的方法;其三、将 PCTL 限界模型检测问题转化为线性方程组的求解问题,而不是传统模型检测方法中的命题公式满足性求解问题.

据我们所知,目前还没有任何工作来探讨 PCTL 的限界模型检测问题.唯一与我们的工作比较接近的是 Penna 等学者在文献[21]中所做的工作.他们在 PCTL 的基础上提出了一种新的时态算子 U^k,得到一种新的时态逻辑 BPCTL,然后针对 U^k 设计了一套算法.直观上,他们将 U^k 的解释限制在长度为 k 的路径上.与我们工作的主要不同点在于,他们仍然在全局空间上设计 U^k 的模型检测算法,并能够直接计算出概率度量,因此本质上不是一种限界模型检测方法.而我们的算法建立在局部空间上,是对真实概率度量的逐步逼近,这遵循限界模型检测的思想.

2 离散时间马尔可夫链与概率计算树逻辑

离散时间马尔可夫链是一簇随机变量$\{X(k)|k=0,1,2,\ldots\}$,这里,$X(k)$是在每一个离散步的观察.$X(k)$的取值称为状态,状态空间的集合是离散的.离散时间马尔可夫链必须满足马尔可夫性质,即 $X(k)$仅仅依赖于 $X(k-1)$,而与

$X(0),\ldots,X(k-2)$无关.另外,我们考虑的离散时间马尔可夫链是齐次的,这意味着状态之间的转换概率独立于时间.因此,给出状态之间的转换概率,就足够描述离散时间马尔可夫链.

定义 1. 离散时间马尔可夫链 $M=(S,P,s_{in},Ap,L)$是一个五元组,这里,

- S 是有限状态集;
- $P:S\times S\to[0,1]$是转换概率函数,且满足对任意的状态 $s\in S$, $\sum_{s'\in S}P(s,s')=1$;
- $s_{in}\in S$ 是初始状态;
- Ap 是有限的原子命题集;
- $L:S\to 2^{Ap}$ 是标记函数.

直觉上,一个离散时间马尔可夫链是一个所有转换关系附上离散概率的 Kripke 结构.在定义概率计算树 PCTL 的语法和语义之前,我们首先回顾一下概率论方面的基本内容.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.集合$\Pi\subseteq 2^{\Omega}$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$\bigcup_{i\geqslant 1}E_i\in\Pi$.

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:1) $Pr(\Omega)=1$;2) 对Π中两两不相交的序列 $E_1,E_2,\ldots$, $Pr(\bigcup_{i=1}^{\infty}E_i)=\bigcup_{i=1}^{\infty}Pr(E_i)$. 称$\Pi$中任何元素是可度量的.

定义 2(路径). 设 M 为离散时间马尔可夫链,M 中的路径π是一个无穷状态序列 $s_0,s_1,\ldots$,使得$\forall i\geqslant 0$, $P(s_i,s_{i+1})>0$.

为表达方便起见,引入记号 $Paths(s)$表示从状态 s 出发的路径集合;对于路径$\pi=s_0,s_1,\ldots$,引入$\pi(i)$表示π上的第 i 个状态 s_i.对于离散时间马尔可夫链 M 和状态 s,令$\Omega=Paths(s)$,Π为σ代数,定义为$\Pi=\{C(\rho)|\rho\in sS^*\}$.这里,$C(\rho)=\{\pi|\rho$是$\pi$的有限前缀$\}$.$\Pi$上的概率度量 Pr_s 定义为 $Pr_s(C(s_1,s_2,\ldots,s_n))=\prod_{1\leqslant i\leqslant n-1}P(s_i,s_{i+1})$. 这里,$s_1=s$.这样,我们从离散时间马尔可夫链 M 和状态 s 演绎出了一个概率空间.

概率计算树逻辑PCTL是基于CTL的分支时态逻辑.PCTL由状态公式和路径公式组成,分别在马尔可夫链的状态上解释状态公式,在路径上解释路径公式.PCTL 与 CTL 的主要区别在于去除了全称和存在路径量词,引入了概率算子 $P_J(\phi)$.这里,ϕ是路径公式,J 是[0,1]上的某个区间.PCTL 的形式化定义如下.

定义 3(概率计算树逻辑 PCTL). 原子命题集 Ap 上的 PCTL 状态公式定义如下:

$$\phi::=\text{true}|a|\phi_1\wedge\phi_2|\phi_1\vee\phi_2|\neg\phi|P_J(\varphi).$$

这里,$a\in Ap$,φ是一条路径公式,$J\subseteq[0,1]$是以有理数为边界的区间(选择有理数作为边界是为了方便计算机处理).PCTL 路径公式定义如下:$\varphi::=X\phi|F\phi|G\phi|\phi_1U\phi_2|\phi_1R\phi_2$.这里,$\phi,\phi_1,\phi_2$ 是状态公式.

为方便起见,不再显式地书写区间,而采用简写方式,例如 $P_{\leqslant 0.5}(\varphi)$表示 $P_{[0,0.5]}(\varphi)$,$P_{=1}(\varphi)$表示 $P_{[1,1]}(\varphi)$.PCTL 的满足性关系定义如下.

定义 4(概率计算树逻辑 PCTL 的满足性关系). 令 $a\in Ap$ 为原子命题,$M=(S,P,s_{in},Ap,L)$是离散时间马尔可夫链,$s\in S$,ϕ_1,ϕ_2 是 PCTL 状态公式,φ是 PCTL 路径公式.对于状态公式满足性关系$\vDash$定义为:

- $s\vDash a$ 当且仅当 $a\in L(s)$;
- $s\vDash\neg\phi_1$ 当且仅当 $s\nvDash\phi_1$;
- $s\vDash\phi_1\wedge\phi_2$ 当且仅当 $s\vDash\phi_1$ 且 $s\vDash\phi_2$;
- $s\vDash\phi_1\vee\phi_2$ 当且仅当 $s\vDash\phi_1$ 或者 $s\vDash\phi_2$;
- $s\vDash P_J(\varphi)$当且仅当 $Pr(s\vDash\varphi)\in J$,这里,$Pr(s\vDash\varphi)=Pr_s(\{\pi\in Paths(s)|\pi\vDash\varphi\})$.

对于 M 中的路径π,满足性关系$\vDash$定义为:

- $\pi\vDash X\phi_1$ 当且仅当$\pi(1)\vDash\phi_1$;

- $\pi\models F\phi_1$ 当且仅当存在自然数 i,使得$\pi(i)\models\phi_1$;
- $\pi\models G\phi_1$ 当且仅当对任意的自然数 i,$\pi(i)\models\phi_1$;
- $\pi\models\phi_1U\phi_2$ 当且仅当存在自然数 j,使得$\pi(j)\models\phi_2$,且对任意小于 j 的自然数 i,$\pi(i)\models\phi_1$;
- $\pi\models\phi_1R\phi_2$ 当且仅当对任意自然数 j,$\pi(j)\models\phi_2$;或者存在自然数 k,使得$\pi(k)\models\phi_1$,且对任意不大于 k 的自然数 i,$\pi(i)\models\phi_2$.

3 PCTL 的限界模型检测

3.1 PCTL的等价性

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PCTL中的计算树逻辑部分,我们可以采用CTL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 $P_{\geqslant p}$ 这类算子,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$P_{\leqslant p}$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,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 PCTL 公式转换为等价的且概率约束为$\geqslant p$ 或者$>p$ 形式的 PCTL 公式.

定义 5(PCTL 公式的等价). 称 PCTL 状态公式ϕ,φ是等价的,记为$\phi\equiv\varphi$,当且仅当对任意的离散时间马尔可夫链 M,任意的 $s\in S$,$s\models\phi$当且仅当 $s\models\varphi$.

不难验证我们有下面的等价关系:

- $P_{\leqslant p}(X\phi)\equiv P_{\geqslant 1-p}(X\neg\phi)$;$P_{<p}(X\phi)\equiv P_{>1-p}(X\neg\phi)$;
- $P_{\leqslant p}(F\phi)\equiv P_{\geqslant 1-p}(G\neg\phi)$;$P_{<p}(F\phi)\equiv P_{>1-p}(G\neg\phi)$;
- $P_{\leqslant p}(G\phi)\equiv P_{\geqslant 1-p}(F\neg\phi)$;$P_{<p}(G\phi)\equiv P_{>1-p}(F\neg\phi)$;
- $P_{\leqslant p}(\phi U\varphi)\equiv P_{\geqslant 1-p}(\neg(\phi U\varphi))\equiv P_{\geqslant 1-p}(\neg\phi R\neg\varphi)$;$P_{<p}(\phi U\varphi)\equiv P_{>1-p}(\neg(\phi U\varphi))\equiv P_{>1-p}(\neg\phi R\neg\varphi)$;
- $P_{\leqslant p}(\phi R\varphi)\equiv P_{\geqslant 1-p}(\neg(\phi R\varphi))\equiv P_{\geqslant 1-p}(\neg\phi U\neg\varphi)$;$P_{<p}(\phi R\varphi)\equiv P_{>1-p}(\neg(\phi R\varphi))\equiv P_{>1-p}(\neg\phi U\neg\varphi)$.

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 PCTL 的表达力:

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

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

3.2 PCTL的限界语义

定义 6(PCTL 的限界语义). 令 $a\in Ap$ 为原子命题,$M=(S,P,s_{in},Ap,L)$是离散时间马尔可夫链,$s\in S,\phi_1,\phi_2$ 是$\text{PCTL}_{\geqslant}$状态公式,φ是 $\text{PCTL}_{\geqslant}$路径公式,k 为自然数(称为界).对于状态公式满足性关系$\models_k$定义为:

- $s\models_k a$ 当且仅当 $a\in L(s)$;
- $s\models_k\phi_1\wedge\phi_2$ 当且仅当 $s\models_k\phi_1$ 且 $s\models_k\phi_2$;
- $s\models_k\phi_1\vee\phi_2$ 当且仅当 $s\models_k\phi_1$ 或者 $s\models_k\phi_2$;
- $s\models_k P_{\geqslant p}(\varphi)$当且仅当 $Pr(s\models_k\varphi)\geqslant p$,这里,$Pr(s\models_k\varphi)=Pr_s(\{\pi\in Paths(s)|\pi\models_k\varphi\})$.

对于 M 中的路径π,满足性关系$\models_k$定义为:

- $\pi\models_k X\phi_1$ 当且仅当 $k\geqslant 1$ 且$\pi(1)\models_k\phi_1$;
- $\pi\models_k F\phi_1$ 当且仅当存在自然数 $i\leqslant k$,使得$\pi(i)\models_k\phi_1$;
- $\pi\models_k G\phi_1$ 当且仅当对任意的自然数 $i\leqslant k$,$\pi(i)\models_k\phi_1$,且存在自然数 $0\leqslant j\leqslant k$,使得 $P(\pi(k),\pi(j))>0$,$\pi=\pi(0)\ldots$

$\pi(j-1)(\pi(j),\ldots,\pi(k))^{\omega}$;

- $\pi\models_k\phi_1U\phi_2$ 当且仅当存在自然数 $j\leqslant k$,使得$\pi(j)\models_k\phi_2$,且对任意小于 j 的自然数 i,$\pi(i)\models_k\phi_1$;
- $\pi\models_k\phi_1R\phi_2$ 当且仅当:1) 对任意的自然数 $i\leqslant k$,$\pi(i)\models_k\phi_2$,且存在自然数 $0\leqslant j\leqslant k$,使得 $P(\pi(k),\pi(j))>0$,$\pi=\pi(0)\ldots\pi(j-1)(\pi(j),\ldots,\pi(k))^{\omega}$;或者,2) 存在自然数 $m\leqslant k$,使得$\pi(m)\models_k\phi_1$,且对任意小于 m 的自然数 i,$\pi(i)\models_k\phi_2$.

定理 1. 令 $a\in Ap$ 为原子命题,$M=(S,P,s_{in},Ap,L)$是离散时间马尔可夫链,$s\in S$,ϕ是 PCTL$_{\geqslant}$状态公式,k 为自然数.如果 $s\models_k\phi$,则 $s\models\phi$.

证明:采用对ϕ的长度进行归纳的方法来证明结论.

Case 1. $\phi=a$

$s\models_k a$ 说明 $a\in L(s)$,由定义 4 中的满足性关系,直接可得 $s\models a$;

Case 2. $\phi=\phi_1\wedge\phi_2$

$s\models_k\phi_1\wedge\phi_2$ 说明 $s\models_k\phi_1$,$s\models_k\phi_2$.由归纳假设可知:$s\models\phi_1$,$s\models\phi_2$,即 $s\models\phi_1\wedge\phi_2$.

Case 3. $\phi=\phi_1\vee\phi_2$

$s\models_k\phi_1\vee\phi_2$ 说明 $s\models_k\phi_1$ 或者 $s\models_k\phi_2$.由归纳假设可知:$s\models\phi_1$ 或者 $s\models\phi_2$,即 $s\models\phi_1\vee\phi_2$.

Case 4. $\phi=P_{\geqslant p}(\varphi)$

$s\models_k P_{\geqslant p}(\varphi)$说明 $Pr(s\models_k\varphi)\geqslant p$,即 $Pr(s\models_k\varphi)=Pr_s(\{\pi\in Paths(s)|\pi\models_k\varphi\})\geqslant p$.对于时态算子 X,F,U,限界语义与文献[15]中定义的一致,因此有$\pi\models_k\varphi$蕴含$\pi\models\varphi$,从而 $Pr_s(\{\pi\in Paths(s)|\pi\models\varphi\})\geqslant Pr_s(\{\pi\in Paths(s)|\pi\models_k\varphi\})\geqslant p$,即 $s\models P_{\geqslant p}(\varphi)$.对于算子 R,$\phi_1R\phi_2\equiv G\phi_2\vee\phi_2U(\phi_1\wedge\phi_2)$,因此只需考察算子 G.

令$\varphi=G\psi$,设$\pi\models_k\varphi$,依据限界语义的定义,对任意自然数 $i\leqslant k$,$\pi(i)\models_k\psi$,由归纳可知:$\pi(i)\models\psi$.又由限界语义的定义可知,存在 $0\leqslant j\leqslant k$,使得 $P(\pi(k),\pi(j))>0$,$\pi=\pi(0)\ldots\pi(j-1)(\pi(j),\ldots,\pi(k))^{\omega}$,则$\forall i\geqslant 0$,$\pi(i)\models\varphi$,即$\pi\models\varphi$.因此,

$$\{\pi\in Paths(s)|\pi\models_k\varphi\}\subseteq\{\pi\in Paths(s)|\pi\models\varphi\},$$

即 $Pr(s\models\varphi)=Pr_s(\{\pi\in Paths(s)|\pi\models\varphi\})\geqslant Pr_s(\{\pi\in Paths(s)|\pi\models_k\varphi\})\geqslant p$. □

定理 1 表明,限界语义的定义是正确的.即,在局部空间中成立的属性在全局空间中也成立.

3.3 限界模型检测过程终止的判断

定理 1 告诉我们,如果存在自然数 k 使得 $s\models_k\phi$,则可推断出 $s\models\phi$.现在的问题是,如果 $s\not\models_k\phi$,则不存在自然数 k 使得 $s\models_k\phi$.换句话讲,当 $s\not\models_k\phi$时,我们面临两种选择:其一是增加 k 的值继续搜索,其二是停止搜索.因此,我们需要一个判别标准来判断当前状态下应该持有的选择.本节我们将探讨这种标准.首先回顾一下分支时态逻辑模型检测中的完全界的概念.

定义 7. 称自然数 CT 是完全界当且仅当如果 $s\models\phi$,则一定存在自然数 $k\leqslant CT$,使得 $s\models_k\phi$.

在知道完全界 CT 的情况下,当 $s\not\models_k\phi$时,如果 k 不大于 CT,则增加 k 的值继续搜索;否则停止搜索,并返回信息:$s\not\models\phi$.对于分支时态逻辑或者线性时态逻辑的限界模型检测,完全界是存在的;但是对于 PCTL 限界模型检测,完全界则不一定存在.

考察一个简单的通信协议[4],该协议的离散时间马尔可夫链如图 1 所示.*start* 是初始状态,且在 *start* 下产生一条消息.消息产生后,系统进入状态 *try*.在 *try* 状态下,消息成功发送的概率为 $\frac{9}{10}$,消息丢失的概率为 $\frac{1}{10}$.并且在消息丢失的情况下,消息会被不断地发送,直至成功为止.发送成功后,系统返回初始状态.

对于状态 s,引入原子命题 a_s 表示当前状态为 s.考察属性 $P_{=1}(Fa_{deliv})$,即消息发送成功的概率为 1.通过计算发现:

$$\begin{aligned}Pr(start\models Fa_{deliv}) &= Pr_{start}\{\pi\in Paths(start)\mid\pi\models Fa_{deliv}\}\\ &= Pr_{start}\{\pi=start,try,(lost,try)^r,deliv,\ldots\mid r\geqslant 0\}\\ &= \frac{9}{10}+\frac{1}{10}\cdot\frac{9}{10}+\frac{1}{10}\cdot\frac{1}{10}\cdot\frac{9}{10}+\frac{1}{10}\cdot\frac{1}{10}\cdot\frac{1}{10}\cdot\frac{9}{10}+\ldots=1,\end{aligned}$$

即 $start\models P_{=1}(Fa_{deliv})$.令 k 为界,则

$$Pr(start\models_k Fa_{deliv})=Pr_{start}\{\pi\in Paths(start)\mid \pi\models_k Fa_{deliv}\}$$
$$=Pr_{start}\left\{\pi=start,try,(lost,try)^r,deliv,...\mid r\leqslant\frac{k}{2}-1\right\}$$
$$=\frac{9}{10}+\frac{1}{10}\cdot\frac{9}{10}+\frac{1}{10}\cdot\frac{1}{10}\cdot\frac{9}{10}+...+\left(\frac{1}{10}\right)^{\left\lfloor\frac{k}{2}-1\right\rfloor}\cdot\frac{9}{10}=1-\left(\frac{1}{10}\right)^{\left\lfloor\frac{k}{2}-1\right\rfloor+1}.$$

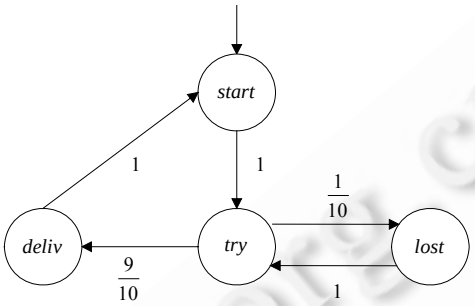

Fig.1 Discrete-Time Markov chain of a simple communication protocol

图 1 一个简单通信协议的离散时间马尔可夫链

比较 $Pr(start\models Fa_{deliv})$和 $Pr(start\models_k Fa_{deliv})$可以发现,对于任意的有限界 k,$Pr(start\models_k Fa_{deliv})<Pr(start\models Fa_{deliv})$.换句话讲,对于属性 $P_{=1}(Fa_{deliv})$,尽管 $start\models P_{=1}(Fa_{deliv})$,但不存在一个有限的自然数 k,使得 $start\models_k P_{=1}(Fa_{deliv})$.因此,下面我们探讨在何种情况下完全界是存在的.

对于 PCTL$_{\geqslant}$路径公式ϕ,满足ϕ的路径是一条无穷的状态演化序列.如果每一条路径均存在一个前缀满足ϕ,则最长前缀序列的长度就是一个完全界.

定理 2. 令 $M=(S,P,s_{in},Ap,L)$为离散时间马尔可夫链,ϕ是 PCTL$_{\geqslant}$路径公式.如果存在从初始状态 s_{in} 出发的有限个状态序列 $seq_1,\ldots,seq_n$,使得对任意满足ϕ的从 s_{in} 出发的路径π,均存在某个自然数 k 使得$\pi\models_k\phi$,且$\pi(0),\ldots,\pi(k)$是 $seq_1,\ldots,seq_n$ 中某个序列的子序列,则一定存在完全界 CT,使得 $s_{in}\models P_{\geqslant r}(\phi)$时 $s_{in}\models_k P_{\geqslant r}(\phi)$,而且 CT 不超过 $seq_1,\ldots,seq_n$ 中最长序列的长度.

定理 2 的证明是直接的,这里不再给出.定理 2 告诉我们,如果对任意满足ϕ的路径均可找到一个有限状态序列对其进行刻画,则完全界不会超过最长有限状态序列的长度.

引理 1. 令 $M=(S,P,s_{in},Ap,L)$是离散时间马尔可夫链,ϕ是 PCTL 状态公式,φ是 PCTL 路径公式,则对于任意的自然数 k,我们有 $s\models_k\phi\rightarrow s\models_{k+1}\phi,\pi\models_k\varphi\rightarrow\pi\models_{k+1}\varphi$.

引理 1 的证明是平凡的,通过对 PCTL 公式的长度进行归纳即可得到,这里省略其证明过程.引理 1 告诉我们,$Pr(s\models_1\phi),Pr(s\models_2\phi),\ldots$是一个递增的序列,且收敛于 $Pr(s\models\phi)$.由收敛的定义可知,对于小于 $Pr(s\models\phi)$的实数 r,必然存在整数 k,使得 $Pr(s\models_k\phi)\geqslant r$,因此有下面的定理.

定理 3. 令 $M=(S,P,s_{in},Ap,L)$为离散时间马尔可夫链,ϕ是 PCTL$_{\geqslant}$路径公式,$p=Pr(s_{in}\models\phi)$,则对于公式 $P_{\geqslant r}(s_{in}\models\phi)$ $(r<p)$,一定存在完全界 CT,使得 $s_{in}\models P_{\geqslant r}(\phi)$时 $s_{in}\models_{CT}P_{\geqslant r}(\phi)$.

证明:由引理 1 可知,$Pr(s_{in}\models_k\phi)$随着 k 的增加而不断增加,且 $\lim\limits_{k\to\infty}Pr(s_{in}\models_k\phi)=Pr(s_{in}\models\phi)$.令$\xi=p-r$,由极限的定义可知,存在自然数 k_ξ,当 $k>k_\xi$时,$|Pr(s_{in}\models_k\phi)-Pr(s_{in}\models\phi)|<\xi$.

此时取 $CT=k_\xi+1$,则$|Pr(s_{in}\models_{CT}\phi)-Pr(s_{in}\models\phi)|<\xi$,即 $Pr(s_{in}\models_{CT}\phi)>p-\xi$,从而有 $s_{in}\models_{CT}P_{\geqslant r}(\phi)$. □

上述两个定理仅仅告诉我们在一些特殊的情形下完全界是存在的,但是对绝大多数情形而言不仅不知道完全界是否存在,更不知道其有多大.因此,需要提出新的搜索过程终止的判别标准.我们引入数值计算方法中牛顿迭代法常用的计算过程终止判别标准,即给定一个预先设置好的非常小的有理数ξ,当相连两次概率度量

计算结果的差控制在ξ内时,计算终止.

具体来讲,PCTL 限界模型检测过程如下.

算法 1. PCTL 限界模型检测.

输入:离散时间马尔可夫链 $M=(S,P,s_{in},Ap,L)$,PCTL 路径公式ϕ,预先设置的终止标准ξ;

输出:$Pr(s_{in}\vDash\phi)$.

Step 1. 将ϕ转换为等价的 PCTL$_{\geqslant}$公式ϕ'

Step 2. 令 $k=1$,计算 $Pr(s_{in}\vDash_0\phi')$, $Pr(s_{in}\vDash_1\phi')$

Step 3.

While $Pr(s_{in}\vDash_k\phi')-Pr(s_{in}\vDash_{k-1}\phi')\geqslant\xi$ do

{令 $k=k+1$,计算 $Pr(s_{in}\vDash_k\phi')$}

Step 4. 输出 $Pr(s_{in}\vDash_k\phi')$

上述过程存在这样一个问题,即如何计算 $Pr(s_{in}\vDash_k\phi')$,在第 3.4 节我们将探讨 $Pr(s_{in}\vDash_k\phi')$的计算问题.

3.4 PCTL的限界模型检测算法

本节我们将探讨如何将 s_{in} 对 PCTL 公式的满足性关系判定问题转换为线性方程组的求解问题.

对于 PCTL$_{\geqslant}$公式ϕ,我们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,均已知 s 是否满足φ.令 $k\geqslant 0$ 为限界模型检测的界,$x(s,\phi,k)=Pr(s\vDash_k\phi)$,$S_{\varphi,k}=\{s\in S|s\vDash_k\varphi\}$.不同的时态算子对应着不同的转换方法,我们分别加以讨论.而对于原子命题及其否定、$\vee$以及$\wedge$算子,因它们直观、简单,故此处忽略.

Case 1. $\phi=X\varphi$

当 $k=0$ 时,$x(s,\phi,0)=0$;当 $k\geqslant 1$ 时, $x(s,\phi,k)=\sum_{s'\in S_{\varphi,k}}P(s,s')$.

Case 2. $\phi=F\varphi$

当 $k=0$ 时:如果 $s\vDash_0\varphi$,则 $x(s,\phi,0)=1$;否则,$x(s,\phi,0)=0$;

当 $k\geqslant 1$ 时:如果 $s\vDash_k\varphi$,则 $x(s,\phi,k)=1$;否则, $x(s,\phi,k)=\sum_{s'\in S}P(s,s')x(s',\phi,k-1)$.

Case 3. $\phi=G\varphi$

对于任意的 $s\notin S_{\varphi,k}$,$x(s,\phi,k)=0$;对于 $s\in S_{\varphi,k}$,我们有:

当 $k=0$ 时,如果 $P(s,s)=1$,则 $x(s,\phi,0)=1$;否则,$x(s,\phi,0)=0$;

当 $k\geqslant 1$ 时, $x(s,\phi,k)=\lim_{n\to\infty}\sum_{i=0}^{k}\sum_{s_0,\ldots,s_k\in S_{\varphi,k}\wedge s=s_0}P(s_0,s_1)..P(s_{i-1},s_i)(P(s_i,s_{i+1})...P(s_{k-1},s_k)P(s_k,s_i))^n$.

事实上,当 $k\geqslant 1$ 时,我们计算的是 $Pr_s\{\pi|s_0=s,\exists s_1,\ldots,s_k\in S_{\varphi,k},\exists 0\leqslant i\leqslant k,\pi=s_0,\ldots,s_{i-1},(s_i,\ldots,s_k)^{\omega}\}$.

Case 4. $\phi=\varphi U\gamma$

当 $k=0$ 时:如果 $s\vDash_0\gamma$,则 $x(s,\phi,0)=1$;否则,$x(s,\phi,0)=0$;

当 $k\geqslant 1$ 时:如果 $s\nvDash_k\varphi$,则 $x(s,\phi,k)=0$;否则, $x(s,\phi,k)=\sum_{s'\in S}P(s,s')x(s',\phi,k-1)$.

Case 5. $\phi=\varphi R\gamma$

当 $k=0$ 时:如果 $s\vDash_0\varphi$,则 $x(s,\phi,0)=1$;否则,$x(s,\phi,0)=0$;

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:

$$x(s,\phi,k)=\lim_{n\to\infty}\sum_{i=0}^{k}\sum_{s_0,\ldots,s_k\in S_{\gamma,k}\cap S_{\neg\varphi,k}}P(s_0,s_1)..P(s_{i-1},s_i)(P(s_i,s_{i+1})...P(s_{k-1},s_k)P(s_k,s_i))^n+x(s,\gamma U(\gamma\wedge\varphi),k).$$

这里,$s_0=s$.对于同时满足 $G\gamma,\gamma U(\gamma\wedge\varphi)$的路径$\pi$,一定存在自然数 i,使得$\pi(i)\vDash_k\varphi$.因此在第 1 部分,令 $s_0,\ldots,s_k\in S_{\gamma,k}\cap S_{\neg\varphi,k}$,可以保证不重复计算 $Pr_s(\{\pi\in Paths(s)|\pi\vDash_k G\gamma\wedge\pi\vDash_k\gamma U\varphi\})$.

现在分析变元数与模型、界、公式大小之间的依赖关系.

定义 8(*l* 步可达). 对于状态 s:1) 如果 $s_0=s$,则称 s_0 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 $P(s_{l-1},s_l)>0$,则称 s_l 是从 s 出发 l 步可达的.

对 PCTL$_{\geqslant}$公式ϕ,令$|\phi|$表示ϕ中出现的符号的数目.设 $M=(S,P,s_{in},Ap,L)$为离散时间马尔可夫链,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,在$\phi=\varphi R\gamma$的情况下计算概率度量时,引入了一个新的公式$\gamma\wedge\varphi$.因此,V 与 $k,N_0,\ldots,N_k,|\phi|$之间的关系为 $V\leqslant(N_0+\ldots+N_k)\times|\phi|\times k\times 2$.

4 实例:IPv4 零配置协议

本节我们将通过一个实际的例子(IPv4 零配置协议)来说明 PCTL 的限界模型检测过程.家庭局部网络与外面的网络都有一个接口来保持通信,这种 Ad Hoc 网必须是热插拔的,且是自我配置的.这意味着当一个新的应用连接到网络时,必须给它自动配置唯一的 IP 地址.IPv4 零配置协议正是为家庭局部网络的应用而设计的,其主要功能是为新的应用动态配置 IP 地址.

IPv4 零配置协议通过下述方式解决 IP 地址的自动配置问题:首先,主机在 65 024 个可用的地址中选择一个(称为 U),并且发布一条消息“谁在使用地址 U?”,如果网络中有其他主机正在使用 U,则其通过消息“我在使用 U”作为回应.在收到回应消息后,主机重新配置 IP 地址,并重复刚才的过程.因为消息会丢失或者主机忙,发布的消息可能不能到达某些主机.为了提高协议的可靠性,主机需要将同一消息发送 n 次,每次间隔 l 个时间单位.因此,主机在发送完 n 次消息并且在 $n\cdot l$ 个时间单位内没有收到回复的消息后,就可以使用选择的地址.但是发送的消息可能会全部丢失,因此执行该协议后,主机仍然可能会使用正在使用的地址.这种情况称为地址冲突,将会导致 TCP/IP 连接失效.

我们研究单个主机试图在网络中配置 IP 地址的行为.假设网络中有 m 个主机.因为有 65 024 个地址可供选择,主机选择一个已经使用的 IP 地址的概率是 $q=m/65024$.假设主机需要将同一消息发送 n 次,并且当主机发送一条包含已经使用的 IP 地址的消息时,其没有收到回应的概率为 p.这里,没有回应包含 3 种情况:发送的消息丢失、响应的主机忙、回应的消息丢失.

现在解释一下如图 2 所示的离散时间马尔可夫链 M.M 有 n+3 个状态(M 中取 n=4):$\{s_{in},s_1,\ldots,s_n,ok,err\}$.在初始化状态 s_{in} 下,主机随机选择一个 IP 地址;在状态 $s_i(1\leqslant i\leqslant n)$下,主机发送它的第 i 个消息;在状态 ok 下,主机成功地选择了一个新的 IP 地址;在状态 err 下,主机选择了一个已经使用的 IP 地址.

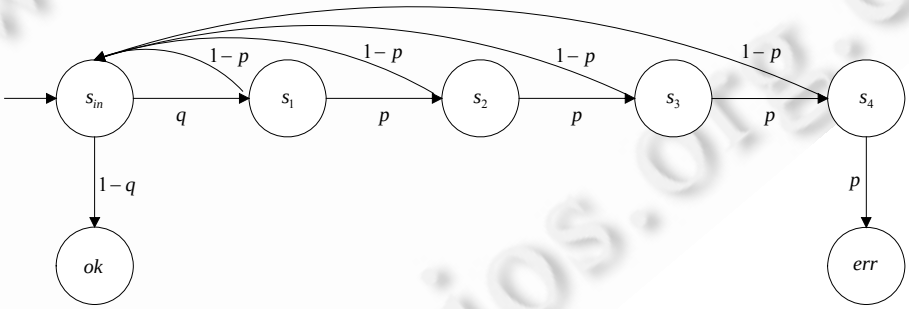

Fig.2 Discrete-Time Markov chain of IPv4 zero configuration protocol

图 2 IPv4 零配置协议的离散时间马尔可夫链

在初始状态 s_{in} 下,有两种可能:选择一个已经使用的 IP 地址的概率为 q,并且转移到状态 s_1;选择一个新的 IP 地址的概率为 $1-q$,并且转移到状态 ok.在状态 $s_i(1\leqslant i<n)$下,主机发布关于选择的 IP 地址的消息,其不能得到其他主机回应的概率为 p,并且继续发布消息,同时转移到状态 s_{i+1};得到其他主机回应的概率为 $1-p$,同时返回到初始状态重新配置 IP 地址.状态 s_n 的行为是类似的,除了如果得不到回应,主机将启用一个已经被使用的 IP 地址,并且进入状态 err 以外.

令原子命题 a_s 表示当前处于状态 s,即 a_s 为真,表示当前状态为 s,我们考虑下面的属性:$P_{\geqslant r}(Fa_{err})$,即主机使用一个已经使用的 IP 地址的概率不低于 r.

取 k=5,由第 3.4 节中的限界模型检测算法可得:

$$x(s_{in},Fa_{err},5)=(1-q)\cdot x(ok,Fa_{err},4)+q\cdot x(s_1,Fa_{err},4).$$

进一步计算可得:

$$x(ok,Fa_{err},4)=0;x(s_1,Fa_{err},4)=p\cdot x(s_2,Fa_{err},3)+(1-p)\cdot x(s_{in},Fa_{err},3).$$

一直继续下去,最后得到的线性方程组如公式(1)所示.在用 matlab7.0 进行求解时,我们取 q=20/65024,p=0.1,最终得出 $x(s_{in},Fa_{err},5)=3.075787\cdot10^{-8}$,即 $Pr(s_{in}\models_k Fa_{err})=3.075787\cdot10^{-8}$.利用文献[4]中介绍的基于不动点运算的模型检测算法,可得 $Pr(s_{in}\models Fa_{err})$的实际值为 $3.075882\cdot10^{-8}$.

$$\begin{cases}
x(s_{in},Fa_{err},5)=(1-q)\cdot x(ok,Fa_{err},4)+q\cdot x(s_1,Fa_{err},4)\\
x(ok,Fa_{err},4)=0\\
x(s_1,Fa_{err},4)=p\cdot x(s_2,Fa_{err},3)+(1-p)\cdot x(s_{in},Fa_{err},3)\\
x(s_2,Fa_{err},3)=p\cdot x(s_3,Fa_{err},2)+(1-p)\cdot x(s_{in},Fa_{err},2)\\
x(s_{in},Fa_{err},3)=(1-q)\cdot x(ok,Fa_{err},2)+q\cdot x(s_1,Fa_{err},2)\\
x(s_3,Fa_{err},2)=p\cdot x(s_4,Fa_{err},1)+(1-p)\cdot x(s_{in},Fa_{err},1)\\
x(s_{in},Fa_{err},2)=(1-q)\cdot x(ok,Fa_{err},1)+q\cdot x(s_1,Fa_{err},1)\\
x(ok,Fa_{err},2)=0\\
x(s_1,Fa_{err},2)=p\cdot x(s_2,Fa_{err},1)+(1-p)\cdot x(s_{in},Fa_{err},1)\\
x(s_4,Fa_{err},1)=p\cdot x(err,Fa_{err},0)+(1-p)\cdot x(s_{in},Fa_{err},0)\\
x(s_{in},Fa_{err},1)=(1-q)\cdot x(ok,Fa_{err},0)+q\cdot x(s_1,Fa_{err},0)\\
x(ok,Fa_{err},1)=0\\
x(s_1,Fa_{err},1)=p\cdot x(s_2,Fa_{err},0)+(1-p)\cdot x(s_{in},Fa_{err},0)\\
x(s_2,Fa_{err},1)=p\cdot x(s_3,Fa_{err},0)+(1-p)\cdot x(s_{in},Fa_{err},0)\\
x(err,Fa_{err},0)=1\\
x(s_{in},Fa_{err},0)=0\\
x(ok,Fa_{err},0)=0\\
x(s_1,Fa_{err},0)=0\\
x(s_2,Fa_{err},0)=0\\
x(s_3,Fa_{err},0)=0
\end{cases}\tag{1}$$

5 实验结果

本节我们通过 3 个测试用例来探讨限界模型检测方法的优缺点以及适用的场景.方程组选用 matlab7.0 求解.

测试用例 1:图 1 所示的一个简单的通信协议,测试的属性为 $P_{\geqslant 0.8}(Fa_{try}),P_{\geqslant 1}(Fa_{try}),P_{\geqslant 0.8}(Fa_{deliv}),P_{\geqslant 1}(Fa_{deliv})$.

测试用例 2:掷骰子赌博游戏[4].

该游戏基于对两个骰子滚动结果的打赌.第 1 次滚动两个骰子的结果决定了是否需要继续滚动骰子.当滚动的结果为 7 或者 11 时,游戏结束,玩家赢.当结果为 2,3 或者 12 时,玩家输.对于其他滚动的结果,需要继续滚动骰子,但是之前掷骰子得到的点数已经被记录下来.如果下一次掷骰子结果为 7 或者为记录的点数,则游戏结束.当结果为 7 时,玩家输,为记录的点数时,玩家赢.在任何其他情况下,滚动骰子直到出现 7 或者记录的点数.游戏的离散时间马尔可夫链如图 3 所示.*start* 是唯一的初始状态.

验证的属性为 $P_{\geqslant 0.24}(\neg(a_8\vee a_9\vee a_{10})Ua_{won})(P_{\geqslant 0.32}(\neg(a_8\vee a_9\vee a_{10})Ua_{won}))$:在滚动结果不出现 8,9,10 的情况下,玩家赢的概率不低于 0.24(0.32).

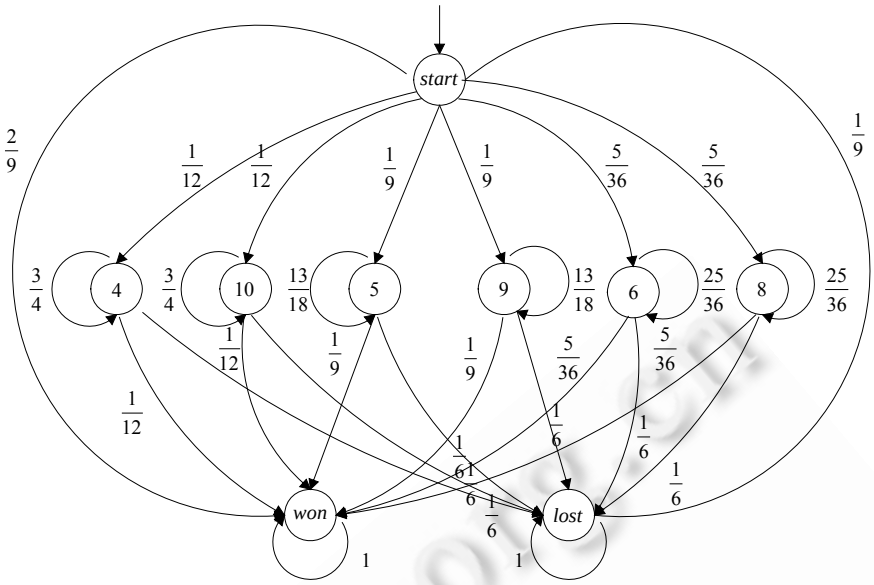

Fig.3 Discrete-Time Markov chain of a craps gambling game

图 3 掷骰子的离散时间马尔可夫链

测试用例 3:IPv4 零配置协议.

验证的属性为 $P_{\geqslant 3.075\cdot 10^{-8}}(Fa_{err}), P_{\geqslant 3.0758\cdot 10^{-8}}(Fa_{err}), P_{\geqslant 3.000\cdot 10^{-5}}(Fa_{s_2}), P_{\geqslant 3.075\cdot 10^{-5}}(Fa_{s_2})$. 我们设置了两种同一消息发送次数的值来改变模型的大小,一种是图 2 所示的 n=4,一种是 n=30.

详细的实验结果见表 1.在表 1 中,令 $\phi_1=Fa_{try}, \phi_2=Fa_{deliv}, \phi_3=\neg(a_8\vee a_9\vee a_{10})Ua_{won}, \phi_4=Fa_{err}, \phi_5=Fa_{s_2}$.在这列中,“无”表示使用的是文献[4]中介绍的无界模型检测算法,变量数是线性方程组中未知数的个数,概率度量是 $Pr(s\vDash_k\phi)$的值(对于无界的情况,概率度量是 $Pr(s\vDash\phi)$的值).真值中,T 表示属性为真,F 表示属性为假.

Table 1 Comparison between the bounded and unbounded model checking

表 1 限界模型检测与无界模型检测方法的比较

测试用例	属性	界	变量数	概率度量	属性	真值	属性	真值
用例 1	ϕ_1	无	4	1	$P_{\geqslant 0.8}(\phi_1)$	T	$P_{\geqslant 1}(\phi_1)$	T
用例 1	ϕ_1	1	2	1	$P_{\geqslant 0.8}(\phi_1)$	T	$P_{\geqslant 1}(\phi_1)$	T
用例 1	ϕ_1	2	2	1	$P_{\geqslant 0.8}(\phi_1)$	T	$P_{\geqslant 1}(\phi_1)$	T
用例 1	ϕ_1	3	2	1	$P_{\geqslant 0.8}(\phi_1)$	T	$P_{\geqslant 1}(\phi_1)$	T
用例 1	ϕ_2	无	4	1	$P_{\geqslant 0.8}(\phi_2)$	T	$P_{\geqslant 1}(\phi_2)$	F
用例 1	ϕ_2	1	2	0	$P_{\geqslant 0.8}(\phi_2)$	F	$P_{\geqslant 1}(\phi_2)$	F
用例 1	ϕ_2	2	4	9/10	$P_{\geqslant 0.8}(\phi_2)$	T	$P_{\geqslant 1}(\phi_2)$	F
用例 1	ϕ_2	3	5	9/10	$P_{\geqslant 0.8}(\phi_2)$	T	$P_{\geqslant 1}(\phi_2)$	F
用例 1	ϕ_2	4	7	99/100	$P_{\geqslant 0.8}(\phi_2)$	T	$P_{\geqslant 1}(\phi_2)$	F
用例 1	ϕ_2	5	8	99/100	$P_{\geqslant 0.8}(\phi_2)$	T	$P_{\geqslant 1}(\phi_2)$	F
用例 1	ϕ_2	6	9	999/1000	$P_{\geqslant 0.8}(\phi_2)$	T	$P_{\geqslant 1}(\phi_2)$	F
用例 2	ϕ_3	无	9	18/55	$P_{\geqslant 0.24}(\phi_3)$	T	$P_{\geqslant 0.32}(\phi_3)$	T
用例 2	ϕ_3	1	9	2/9	$P_{\geqslant 0.24}(\phi_3)$	F	$P_{\geqslant 0.32}(\phi_3)$	F
用例 2	ϕ_3	2	11	0.246566	$P_{\geqslant 0.24}(\phi_3)$	T	$P_{\geqslant 0.32}(\phi_3)$	F
用例 2	ϕ_3	3	12	0.288323	$P_{\geqslant 0.24}(\phi_3)$	T	$P_{\geqslant 0.32}(\phi_3)$	F
用例 2	ϕ_3	4	13	0.307972	$P_{\geqslant 0.24}(\phi_3)$	T	$P_{\geqslant 0.32}(\phi_3)$	F
用例 2	ϕ_3	5	14	0.322012	$P_{\geqslant 0.24}(\phi_3)$	T	$P_{\geqslant 0.32}(\phi_3)$	T
用例 3 (n=4)	ϕ_4	无	7	$3.075882\cdot 10^{-8}$	$P_{\geqslant 3.075\cdot 10^{-8}}(\phi_4)$	T	$P_{\geqslant 3.0758\cdot 10^{-8}}(\phi_4)$	T
用例 3 (n=4)	ϕ_4	1	3	0	$P_{\geqslant 3.075\cdot 10^{-8}}(\phi_4)$	F	$P_{\geqslant 3.0758\cdot 10^{-8}}(\phi_4)$	F
用例 3 (n=4)	ϕ_4	2	5	0	$P_{\geqslant 3.075\cdot 10^{-8}}(\phi_4)$	F	$P_{\geqslant 3.0758\cdot 10^{-8}}(\phi_4)$	F

Table 1 Comparison between the bounded and unbounded model checking (continue)

表 1 限界模型检测与无界模型检测方法的比较(续)

测试用例	属性	界	变量数	概率度量	属性	真值	属性	真值
用例 3 (n=4)	ϕ_4	3	9	0	$P_{\geqslant 3.075\cdot 10^{-8}}(\phi_4)$	F	$P_{\geqslant 3.0758\cdot 10^{-8}}(\phi_4)$	F
用例 3 (n=4)	ϕ_4	4	12	0	$P_{\geqslant 3.075\cdot 10^{-8}}(\phi_4)$	F	$P_{\geqslant 3.0758\cdot 10^{-8}}(\phi_4)$	F
用例 3 (n=4)	ϕ_4	5	20	$3.075787\cdot 10^{-8}$	$P_{\geqslant 3.075\cdot 10^{-8}}(\phi_4)$	T	$P_{\geqslant 3.0758\cdot 10^{-8}}(\phi_4)$	F
用例 3 (n=30)	ϕ_5	无	33	$3.076639\cdot 10^{-5}$	$P_{\geqslant 3.000\cdot 10^{-5}}(\phi_5)$	T	$P_{\geqslant 3.075\cdot 10^{-5}}(\phi_5)$	T
用例 3 (n=30)	ϕ_5	1	3	0	$P_{\geqslant 3.000\cdot 10^{-5}}(\phi_5)$	F	$P_{\geqslant 3.075\cdot 10^{-5}}(\phi_5)$	F
用例 3 (n=30)	ϕ_5	2	5	$3.075787\cdot 10^{-5}$	$P_{\geqslant 3.000\cdot 10^{-5}}(\phi_5)$	T	$P_{\geqslant 3.075\cdot 10^{-5}}(\phi_5)$	T
用例 3 (n=30)	ϕ_5	3	9	$3.075787\cdot 10^{-5}$	$P_{\geqslant 3.000\cdot 10^{-5}}(\phi_5)$	T	$P_{\geqslant 3.075\cdot 10^{-5}}(\phi_5)$	T
用例 3 (n=30)	ϕ_5	4	12	$3.076638\cdot 10^{-5}$	$P_{\geqslant 3.000\cdot 10^{-5}}(\phi_5)$	T	$P_{\geqslant 3.075\cdot 10^{-5}}(\phi_5)$	T
用例 3 (n=30)	ϕ_5	5	20	$3.076638\cdot 10^{-5}$	$P_{\geqslant 3.000\cdot 10^{-5}}(\phi_5)$	T	$P_{\geqslant 3.075\cdot 10^{-5}}(\phi_5)$	T

我们以变量数作为衡量一种算法所需时间和空间的指标,即,变量数越多,我们认为该算法所需要的时间和空间越多.从表 1 中我们可以得出下面几个结论:

结论 1. 界越长,限界模型检测得到的概率度量越来越逼近真实的概率度量.

结论 2. 限界模型检测是一种前向搜索状态空间的方法,在属性为真的证据比较短的情况下,能够快速地验证属性.特别是对于如下情况:设ϕ为 PCTL 公式,$Pr(s\models\phi)=p,r<p$,此时,验证 $P_{\geqslant r}(\phi)$往往比较快.

结论 3. 限界模型检测得到的概率度量只能越来越逼近真实的概率度量,但可能永远无法达到,例如对测试用例 1 中的 Fa_{deliv} 属性.一般对于如下情况,限界模型检测往往会失效:设ϕ为 PCTL 公式,$Pr(s\models\phi)=p$,此时,验证 $P_{\geqslant p}(\phi)$可能会失效.

6 限界模型检测过程终止判断标准的修正

首先回顾一下算法 1 中使用的终止标准:给定非常小的有理数ξ,当相连两次概率度量结果的差控制在ξ以内时(即 $Pr(s_{in}\models_k\phi)-Pr(s_{in}\models_{k-1}\phi)\leqslant\xi$),计算过程终止.对于一个数值序列 $x_0,x_1,\ldots$,如果对任意的自然数 i,$|x_{i+1}-x_i|<|x_{i+2}-x_{i+1}|$,则这个标准是有效的.但对于我们使用限界模型检测算法得到的概率度量序列,它是一个递增序列,但不是严格递增的,即可能存在自然数 k,使得 $Pr(s_{in}\models_k\phi)=Pr(s_{in}\models_{k-1}\phi)$.例如在测试用例 1 中,测试属性$\phi_2$ 时,$Pr(s_{in}\models_2\phi_2)=Pr(s_{in}\models_3\phi_2)=\frac{9}{10}$;在测试用例 3($n$=4)中,测试属性$\phi_4$ 时,$Pr(s_{in}\models_1\phi_4)=\ldots=Pr(s_{in}\models_4\phi_4)=0$.对上述两个用例,按照算法 1 的终止标准,我们得到的近似概率度量与真实的概率度量误差还是比较大的,因此有必要对终止标准加以修正.

修正方案 1:比较不相连的两次限界模型检测得出的概率度量.

设 m,k 为自然数,在第 k 步,如果 $Pr(s_{in}\models_k\phi)-Pr(s_{in}\models_{k-m}\phi)\leqslant\xi$,则检测过程终止.具体过程如算法 2 所示.

算法 2. PCTL 限界模型检测(以修正方案 1 为终止标准).

输入:离散时间马尔可夫链 $M=(S,P,s_{in},Ap,L)$,PCTL 路径公式ϕ,预先设置的终止标准ξ,m 为自然数;

输出:$Pr(s_{in}\models\phi)$.

Step 1. 将ϕ转换为等价的 PCTL$_{\geqslant}$公式ϕ'

Step 2. 计算 $Pr(s_{in}\models_0\phi'),\ldots,Pr(s_{in}\models_m\phi')$,令 $k=m$

Step 3.

While $Pr(s_{in}\models_k\phi')-Pr(s_{in}\models_{k-m}\phi')\geqslant\xi$ do

{令 $k=k+1$,计算 $Pr(s_{in}\models_k\phi')$}

Step 4. 输出 $Pr(s_{in}\models_k\phi')$

在算法 2 中取 m=2,可避免测试用例 1 中检测属性ϕ_2 时存在的收敛问题,但不能解决测试用例 3(n=4)中检

测属性ϕ_4的收敛问题;而取 m=4,两个问题都可解决.序列 $Pr(s_{in}\vDash_1\phi)$,$Pr(s_{in}\vDash_2\phi)$,…是一个非严格递增的递增序列,因此在理论上,m的值越大越好.这种方案的主要缺点在于,m的值需要事先给定,而且无法确定最合理的 m 的值.

修正方案 2:比较相连限界模型检测得出的概率度量的差.

设 k 为自然数,在第 k 步,如果$|Pr(s_{in}\vDash_k\phi)-Pr(s_{in}\vDash_{k-1}\phi)|<|Pr(s_{in}\vDash_{k-1}\phi)-Pr(s_{in}\vDash_{k-2}\phi)|$且 $Pr(s_{in}\vDash_k\phi)-Pr(s_{in}\vDash_{k-1}\phi)\leqslant\xi$,则检测过程终止.具体过程如算法 3 所示.

算法 3. PCTL 限界模型检测(以修正方案 2 为终止标准).

输入:离散时间马尔可夫链 $M=(S,P,s_{in},Ap,L)$,PCTL 路径公式ϕ,预先设置的终止标准ξ;

输出:$Pr(s_{in}\vDash\phi)$.

Step 1. 将ϕ转换为等价的 PCTL$_{\geqslant}$公式ϕ'

Step 2. 计算 $Pr(s_{in}\vDash_0\phi')$,$Pr(s_{in}\vDash_1\phi')$,$Pr(s_{in}\vDash_2\phi')$,令 k=2

Step 3.

While $\neg(Pr(s_{in}\vDash_k\phi')-Pr(s_{in}\vDash_{k-1}\phi')\leqslant\xi\wedge|Pr(s_{in}\vDash_k\phi')-Pr(s_{in}\vDash_{k-1}\phi')|<|Pr(s_{in}\vDash_{k-1}\phi')-Pr(s_{in}\vDash_{k-2}\phi')|)$ do

{令 $k=k+1$,计算 $Pr(s_{in}\vDash_k\phi')$}

Step 4. 输出 $Pr(s_{in}\vDash_k\phi')$

算法 3 可以有效地避免测试用例 1 中检测属性ϕ_2和测试用例 3(n=4)中检测属性ϕ_4存在的收敛问题.但是,对于有理数序列 0,0,0.5,0.6,0.65,1,1,1,…,如果取ξ=0.11,则算法 3 得到的近似概率度量为 0.65,与实际的概率度量 1 误差较大.概率计算树逻辑限界模型检测何时终止,依赖于马尔可夫链的结构、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.

7 结 论

为了克服概率计算树模型检测中的状态空间爆炸问题,本文将限界模型检测技术应用到概率计算树模型检测的空间简化上来.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,特别是在终止判别标准的设计与限界模型检测算法方面,解决方案的思想完全有别于传统限界检测技术.进一步地,通过 3 个测试用例说明了限界模型检测在属性为真的证据比较短的情况下,能够快速地验证属性,而且需求的空间比无界模型检测技术要少.在 3 个用例的测试中,我们都是手工计算出线性方程组,因此,下一步的主要工作是开发验证工具,能够完成模型的输入,以及依据不同的界,计算线性方程组.同时,挖掘马尔可夫链的结构、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] Alur R. Model checking: From tools to theory. Lecture Notes in Computer Science, 2008,5000:89−106. [doi: 10.1007/978-3-540-69850-0_6]

[2] Lin HM, Zhang WH. Model checking: Theories, techniques and applications. Acta Electronica Sinica, 2002,30(12A):1907−1912 (in Chinese with English abstract).

[3] Hansson H, Jonsson B. A logic for reasoning about time and reliability. Formal Aspects of Computing, 1994,6(5):512−535. [doi: 10.1007/BF01211866]

[4] Baier C, Katoen JP. Principles of Model Checking. Cambridge: MIT Press, 2008. 745−907.

[5] Kattenbelt M, Kwiatkowska M, Norman G, Parker D. A game-based abstraction-refinement framework for Markov decision processes. Formal Methods in System Design, 2010,36(3):246−280. [doi: 10.1007/s10703-010-0097-6]

[6] Clarke EM. My 27-year quest to overcome the state explosion problem. In: Pitts A, ed. Proc. of the 24th Annual IEEE Symp. on Logic in Computer Science. Washington: IEEE Computer Society, 2009. 3. [doi: 10.1109/LICS.2009.42]

[7] Clarke EM, Grumberg O, Jha S, Lu Y, Veith H. Progress on the state explosion problem in model checking. Lecture Notes in Computer Science, 2001,2000:176−194. [doi: 10.1007/3-540-44577-3_12]

[8] Bryant RE. Graph-Based algorithms for boolean function manipulation. IEEE Trans. on Computers, 1986,35(8):687−691. [doi: 10.1109/TC.1986.1676819]

[9] Burch JR, Clarke EM, McMillan KL. Symbolic model checking: 10^{20} states and beyond. Information and Computation, 1992,98(2): 142−170. [doi: 10.1016/0890-5401(92)90017-A]

[10] Su KL, Luo XY, Lu GF. Symbolic model checking for CTL. Chinese Journal of Computers, 2005,28(11):1798−1806 (in Chinese with English abstract).

[11] Qu WX, Li T, Guo Y, Yang XD. Advances in predicate abstraction. Journal of Software, 2008,19(1):27−38 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/19/27.htm [doi: 10.3724/SP.J.1001.2008.00027]

[12] Wolper P, Godefroid P. Partial order methods for temporal verification. Lecture Notes in Computer Science, 1993,715:233−246. [doi: 10.1007/3-540-57208-2_17]

[13] Emerson EA, Sistla AP. Symmetry and model checking. Formal Methods in System Design, 1996,9(1):105−131. [doi: 10.1007/BF00625970]

[14] Pasareanu CS, Dwyer MB, Huth M. Assume-Guarantee model checking of software: A comparative case study. Lecture Notes in Computer Science, 1999,1680:168−183. [doi: 10.1007/3-540-48234-2_14]

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

[16] Yang JJ, Su KL, Luo XY, Lin H, Xiao YY. Optimization of bounded model checking. Journal of Software, 2009,20(8):2005−2014 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/3387.htm [doi: 10.3724/SP.J.1001.2009.03387]

[17] 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]

[18] Zhang WH. Model checking with SAT-based characterization of ACTL formulas. Lecture Notes in Computer Science, 2007,4789: 191−211. [doi: 10.1007/978-3-540-76650-6_12]

[19] Zhang WH. Bounded semantics of CTL and SAT-based verification. Lecture Notes in Computer Science, 2009,5885:286−305. [doi: 10.1007/978-3-642-10373-5_15]

[20] Xu L, Chen W, Xu YY, Zhang WH. Improved bounded model checking for universal fragment of CTL. Journal of Computer Science and Technology, 2009,24(1):96−109. [doi: 10.1007/s11390-009-9208-5]

[21] Penna GD, Intrigila B, Melatti I, Tronci E, Zilli MV. Bounded probabilistic model checking with the Murφ verifier. Lecture Notes in Computer Science, 2004,3312:214−229. [doi: 10.1007/978-3-540-30494-4_16]

附中文参考文献:

[2] 林惠民,张文辉.模型检测:理论、方法与应用.电子学报,2002,30(12A):1907−1912.

[10] 苏开乐,骆翔宇,吕关锋.符号化模型检测 CTL.计算机学报,2005,28(11):1798−1806.

[11] 屈婉霞,李暾,郭阳,杨晓东.谓词抽象技术研究.软件学报,2008,19(1):27−38. http://www.jos.org.cn/1000-9825/19/27.htm [doi: 10.3724/SP.J.1001.2008.00027]

[16] 杨晋吉,苏开乐,骆翔宇,林瀚,肖茵茵.有界模型检测的优化.软件学报,2009,20(8):2005−2014. http://www.jos.org.cn/1000-9825/3387.htm [doi: 10.3724/SP.J.1001.2009.03387]

[17] 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.

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