中国科学: 信息科学 2016 年 第 46 卷 第 11 期: 1591–1607
SCIENTIA SINICA Informationis
《中国科学》杂志社
SCIENCE CHINA PRESS
不确定信息网络专题

基于决策过程的广义可能性计算树逻辑模型检测

马占有[①②], 李永明[①*]

① 陕西师范大学计算机科学学院, 西安 710119
② 北方民族大学计算机科学与工程学院, 银川 750021
* 通信作者. E-mail: liyongm@snnu.edu.cn

收稿日期: 2016–04–24; 接受日期: 2016–08–02; 网络出版日期: 2016–11–09
国家自然科学基金 (批准号: 11271237, 61228305, 61462001) 和高等学校博士学科点专项科研基金 (批准号: 20130202110001, 20130202120002)

摘要 本文研究了广义可能性计算树逻辑模型检测算法及其在系统验证中的应用, 特别是在非确定性系统验证中的应用. 首先引入作为系统模型的广义可能性决策过程和描述系统属性的广义可能性计算树逻辑, 然后给出基于广义可能性决策过程的广义可能性计算树逻辑模型检测算法. 该算法最大的优点是利用决策过程中的调度, 将模型检测问题转换为多项式时间内模糊矩阵的运算或模糊矩阵不动点的计算. 最后通过一个实例说明了广义可能性计算树逻辑模型检测在非确定性系统中的应用.

关键词 非确定性系统 广义可能性决策过程 调度 广义可能性计算树逻辑 模型检测

1 引言

模型检测 (model checking) [1,2] 是一种重要的形式化验证方法, 主要包括计算树逻辑 (computation tree logic, CTL) 模型检测, 线性时态逻辑 (linear temporal logic, LTL) 模型检测等. 由于模型检测过程是自动完成的, 已被应用到计算机软硬件系统、通信协议、安全协议等方面的分析与验证中, 取得了令人瞩目的成功.

模型检测的基本思想是用状态转换系统 M 表示系统的模型, 用时态逻辑公式 F 描述系统的性质, 这样检测一个系统是否满足一个规范就转换为验证 $M \models F$ 是否成立的模型检测问题. 因此, 经典的模型检测方法是一种定性的研究方法. 然而, 计算机系统正变得日益庞大和复杂, 很多实际的系统被赋予量化行为特征. 例如, 多智能体系统 [3~6] 具有复杂的动态结构及行为特征, 需要人为地增加量化信息来刻画其动态行为特征. 为了处理具有量化信息的系统的验证问题, 定量的模型检测方法引起了学术界和工业界的关注. 如 Hart 等提出了基于概率测度的概率模型检测 [7,8], 用 Markov 链或 Markov 决策过程描述系统的行为, 用概率计算树逻辑或概率线性时态逻辑描述系统的性质. Sultan 等提出了概率多智能体模型检测 [9,10]. Chechik 等研究了取值于有限 De Morgan 代数的多值 Kripke 结构上的 CTL 和 LTL 的模型检测问题 [11,12]. 最近, Pan 等从任意模糊逻辑角度, 讨论了 CTL 模型检测问

引用格式: 马占有, 李永明. 基于决策过程的广义可能性计算树逻辑模型检测. 中国科学: 信息科学, 2016, 46: 1591–1607, doi: 10.1360/N112016-00108

题 [13]. 另外, 他们也研究了取值于任意有限格的 CTL 的模型检测问题 [14]. Li 等将模糊理论中的可能性测度与模型检测技术结合起来, 提出了基于可能性测度的模型检测技术 [15~17]. 可能性模型检测技术主要用于不完备信息的系统和非可加性系统的模型检测 [18~22], 在可能性模型检测技术中, 常用可能性 Kripke 结构 [15,16] 和广义可能性 Kripke 结构 [17] 来描述系统的模型, 其结构特征是状态转换关系满足一定的可能性分布. 换句话说, 在可能性 Kripke 结构和广义可能性 Kripke 结构中, 下一个状态的选择是通过可能性选择, 而不是通过动作的非确定性选择.

在实际系统设计时, 经常会遇到所建模型的非确定性系统, 它的每个状态转换到其他状态有多种可能性分布. 例如具有不完备信息的分布式系统, 由于此类系统包含分布式进程的交替动作, 不能用广义可能性 Kripke 结构完整地描述它们的行为. 为了解决这一问题, 本文引入广义可能性决策过程. 广义可能性决策过程的结构特征是状态转换关系既有动作的非确定性选择, 又有可能性分布. 简单地说, 在广义可能性决策过程中, 一旦非确定的通过动作选定可能性分布, 下一个状态的选择与广义可能性 Kripke 一样, 也是通过可能性选择. 在广义可能性决策过程中, 用非确定性描述分布式系统的并发进程的交替, 可能性分布描述系统的不完备信息. 因此, 广义可能性决策过程非常适合作为具有不完备信息的分布式系统的模型. 本文将研究这类系统的 CTL 模型检测问题.

本文是文献 [17] 研究工作的继续. 针对非确定性系统的验证问题, 首先给出描述此类系统模型的广义可能性决策过程的定义, 然后给出广义可能性决策过程下广义计算树逻辑的语义, 最后研究广义计算树逻辑的模型检测问题. 另外, 我们将本文所获得的研究结果应用到疾病诊断系统的验证.

与我们工作最相关的是 Markov 决策过程上的概率计算树逻辑 (probabilistic CTL, PCTL) [1] 模型检测. 不同点有: (1) 时态逻辑的语法结构不同. PCTL 用概率算子 $\mathbb{P}_J(J \subseteq [0,1])$ 取代 CTL 的路径全称量词 ($\forall$) 和路径存在量词 ($\exists$), 而本文提出的 GPoCTL 用可能性算子 GPo 取代 CTL 的路径全称量词和路径存在量词. (2) 语义模型不同. Markov 决策过程的转移上的权重反映出事件出现的频率, 而广义可能性决策过程的转移上的权重反映出到达目标状态的可能性; 在 Markov 决策过程中, 从同一状态出发的转移上的权重之和应该是 1, 而广义可能性决策过程没有这一约束条件; Markov 决策过程上的标签函数是分明的, 而广义可能性决策过程上的标签函数可以是模糊的. (3) 模型检测算法实现方式不同, PCTL 模型检测采用迭代求线性方程来实现, 而 GPoCTL 模型检测算法是利用模糊矩阵的合成运算.

2 预备知识

为了叙述方便, 本文中用 $\mathbb{N}$ 表示自然数, I 表示指标集, c 表示补运算, 对 $F \subseteq [0,1]$, 集合 F 的最小上界和最大下界分别用 $\bigvee F$ 和 $\bigwedge F$ 表示. $|X|$ 表示集合 X 中元素的个数.

定义1 ([18~24]) 设 X 是一个非空集合, Ω 是以 X 中的一些子集为元素构成的集合. 称 Ω 为 σ-代数, 如果它对可数并及取补集运算封闭。σ- 代数 Ω 上的可能性测度是一个映射 $\mathrm{POS}: \Omega \to [0,1]$, 满足如下条件:

(1) $\mathrm{POS}(\emptyset) = 0$;

(2) $\mathrm{POS}(X) = 1$;

(3) 若 $E_i \in \Omega$, $i \in I$, 则 $\mathrm{POS}(\bigcup_{i\in I} E_i) = \bigvee_{i\in I} \mathrm{POS}(E_i)$.

如果只满足以上条件 (1) 和 (3), 则称 POS 为广义可能性测度. 注意到如果 POS 是幂集 2^X 上的广义可能性测度, 对于任意 $A \subseteq X$, 有 $\mathrm{POS}(A) = \bigvee_{a\in A} \mathrm{POS}(\{a\})$.

定义2 ([17]) 一个广义可能性 Kripke 结构 (generalized possibilistic Kripke structure, GPKS) 是一个五元组 $M=(S,P,I,\mathrm{AP},L)$, 其中

(1) S 是一个可数非空状态集合;

(2) $P:S\times S\to[0,1]$ 是可能性转移分布, 对于任意状态 s, 存在状态 t, 使得 $P(s,t)>0$;

(3) $I:S\to[0,1]$ 是可能性初始分布且存在状态 s, 使得 $I(s)>0$;

(4) AP 是一组原子命题集合;

(5) $L:S\times\mathrm{AP}\to[0,1]$ 是标签函数, $L(s,a)$ 表示原子命题 a 在状态 s 上成立的真值.

对于 GPKS M, 其路径定义为无穷状态序列 $\pi=s_0s_1s_2\cdots\in S^\omega$, 满足对于任意的 i, $P(s_i,s_{i+1})>0$. 令 $\mathrm{Paths}(s)$ 和 $\mathrm{Paths}_{\mathrm{fin}}(s)$ 分别表示 M 中从状态 s 出发所有的无穷路径和有穷路径的集合. 用 $\mathrm{Paths}(M)$ 表示 M 中的所有无穷路径的集合, $\mathrm{Paths}_{\mathrm{fin}}(M)$ 表示 M 中所有形如 $\hat{\pi}=s_0s_1\cdots s_n$ 有穷路径的集合.

3 广义可能性决策过程

本文提出类似于 Markov 决策过程的广义可能性决策过程作为非确定性系统的模型, 具体定义如下.

定义3 广义可能性决策过程 (generalized possibilistic decision process, GPDP) 是一个六元组 $M=(S,\mathrm{Act},P,I,\mathrm{AP},L)$, 其中

(1) S 是一个可数非空状态集合;

(2) Act 是动作的集合;

(3) $P:S\times\mathrm{Act}\times S\to[0,1]$ 是可能性转移分布, 对于任意状态 $s\in S$ 和动作 $\alpha\in\mathrm{Act}$, 存在状态 $t\in S$, 使得 $P(s,\alpha,t)>0$;

(4) $I:S\to[0,1]$ 是可能性初始分布, 存在状态 s 使得 $I(s)>0$;

(5) AP 是一组原子命题集合;

(6) $L:S\times\mathrm{AP}\to[0,1]$ 是标签函数, $L(s,a)$ 表示原子命题 a 在状态 s 上成立的真值.

若 $|S|$, $|\mathrm{Act}|$ 和 $|\mathrm{AP}|$ 都是有穷的, 称 M 为有穷的, 其中 $P(s,\alpha,t)$ 表示状态 s 在动作 α 的作用下到达状态 t 的可能性. 如果存在一个状态 $t\in S$, 使得 $P(s,\alpha,t)>0$, 则称 α 在状态 s 上是可触发的, $\mathrm{Act}(s)$ 表示状态 s 所有可触发动作集合. 若 $P(s,\alpha,t)>0$, 则称 t 是 s 后继. 后文用 $\mathrm{Post}(s,\alpha)=\{t\in S\,|P(s,\alpha,t)>0\}$ 表示状态 s 的所有 α 后继, 用 $\mathrm{Pref}(t)=\{(s,\alpha)\in S\times\mathrm{Act}|P(s,\alpha,t)>0\}$ 表示状态 t 的所有前驱. $P(s,\alpha,T)$ 表示从 s 出发在 α 的作用下到 T 中状态的可能性, 即 $P(s,\alpha,T)=\bigvee_{t\in T}P(s,\alpha,t)$.

对于 GPDP $M=(S,\mathrm{Act},P,I,\mathrm{AP},L)$, 用序列 $\pi=s_0\alpha_0s_1\alpha_1s_2\cdots\in(S\times\mathrm{Act})^\omega$ 表示 M 中的无穷路径, $\mathrm{Paths}(s)$ 和 $\mathrm{Paths}_{\mathrm{fin}}(s)$ 分别表示 M 中从状态 s 出发所有的无穷路径和有穷路径的集合. $\mathrm{Paths}(M)$ 表示 M 中的所有无穷路径的集合. $\mathrm{Paths}_{\mathrm{fin}}(M)$ 表示 M 中所有有穷路径的集合. 本文研究的 GPDP 都是有穷的且对于任意 $s\in S$, $\mathrm{Act}(s)\neq\emptyset$.

注 2.1 有穷 GPKS 是有穷 GPDP 的一种特殊情形, 因为任意的有穷 GPKS 都可以看成状态上只有一个可用动作的有穷 GPDP. 对任意 $s,t\in S$, $\alpha\in\mathrm{Act}(s)$, $a\in\mathrm{AP}$, 如果 $P(s,a,t)=\{0,1\}$ 且 $L(s,a)=\{0,1\}$ 时, 则有穷 GPDP 是经典的状态转换系统 [1].

注 2.2 在 GPDP $M=(S,\mathrm{Act},P,I,\mathrm{AP},L)$ 中, 对于任意 $\alpha\in\mathrm{Act}$, 可能性分布函数 $P:S\times$

$\alpha \times S \to [0,1]$ 可以用模糊矩阵表示. 为方便起见, 记该模糊矩阵为 P_α, 即 $P_\alpha = (P(s,\alpha,t))_{s,t\in S}$, 也称为动作 α 对应的 M 的模糊转移矩阵. 用模糊矩阵 $P_{\max} = \bigvee_{i=0}^{n} P_{\alpha_i}$ 表示 $P : S \times \mathrm{Act} \times S \to [0,1]$ 对应的最大可能性转移矩阵, 即 $(P_{\max}(s,t))_{s,t\in S} = (\bigvee_{\alpha\in \mathrm{Act}(s)} P(s,\alpha,t))_{s,t\in S}$. 用模糊矩阵 $P_{\min} = \bigwedge_{i=0}^{n} P_{\alpha_i}$ 表示 $P : S \times \mathrm{Act} \times S \to [0,1]$ 对应的最小可能性转移矩阵, 即 $(P_{\min}(s,t))_{s,t\in S} = (\bigwedge_{\alpha\in \mathrm{Act}(s)} P(s,\alpha,t))_{s,t\in S}$. 由矩阵 $P_{\max}$ 和矩阵 $P_{\min}$ 可以分别构造 GPKS $M_{\max} = (S, P_{\max}, I, \mathrm{AP}, L)$ 和 GPKS $M_{\min} = (S, P_{\min}, I, \mathrm{AP}, L)$.

对模糊矩阵 $P_{\max}$, 其转移矩阵闭包记为 $P^{+}_{\max}$. 当 S 为有限集, 则 $P^{+}_{\max}$=$P_{\max} \bigvee P^{2}_{\max} \bigvee P^{3}_{\max} \bigvee \cdots \bigvee P^{|S|}_{\max}$, 其中 $P^{k+1}_{\max} = P^{k}_{\max} \circ P_{\max}$, $\circ$ 表示模糊矩阵的取大取小复合运算. 对模糊矩阵 $P_{\max}$, 其自反转移矩阵闭包 $P^{*}_{\max}$ 定义为 $P^{*}_{\max} = P^{0}_{\max} \bigvee P^{+}_{\max}$, 其中 $P^{0}_{\max}$ 表示恒等矩阵. 对于 GPKS $M_{\max} = (S, P_{\max}, I, \mathrm{AP}, L)$, 利用 $P^{+}_{\max}$ 和 $P^{*}_{\max}$ 可以得到两个 GPKS $M^{+}_{\max} = (S, P^{+}_{\max}, I, \mathrm{AP}, L)$ 和 $M^{+}_{\max} = (S, P^{+}_{\max}, I, \mathrm{AP}, L)$. 对 GPKS $M_{\min} = (S, P_{\min}, I, \mathrm{AP}, L)$, 我们可以定义 $P^{+}_{\min}$ 和 $P^{*}_{\min}$ 并构造两个 GPKS $M^{+}_{\min} = \big(S, P^{+}_{\min}, I, \mathrm{AP}, L\big)$ 和 $M^{*}_{\min} = \big(S, P^{+}_{\min}, I, \mathrm{AP}, L\big)$.

例 2.1 图 1 描述的是一个 4 个状态的 GPDP$M = (S, P, \mathrm{Act}, I, \mathrm{AP}, L)$, 其中圆圈表示状态, 圆圈外的符号表示状态名, 圆圈内的符号表示原子命题在状态上的真值, 带标签的弧线表示转移, 有输入箭头的圆圈是初始状态. 则 M 的状态空间 $S = \{S_0, S_1, S_2, S_3\}$, 动作集合 $\mathrm{Act} = \{\alpha, \beta\}$, 以 $S_0 \to S_1 \to S_2 \to S_3$ 的顺序分别给出下列模糊矩阵:

$$I = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad P_\alpha = \begin{pmatrix} 0 & 0 & 0.7 & 0 \\ 0.5 & 0 & 0 & 1 \\ 0 & 0 & 0.8 & 0 \\ 0 & 0 & 1 & 0.4 \end{pmatrix}, \quad P_\beta = \begin{pmatrix} 0 & 0.8 & 0.9 & 0.5 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0.7 & 0 & 0 & 0 \end{pmatrix},$$

$$P_{\max} = \begin{pmatrix} 0 & 0.8 & 0.9 & 0.5 \\ 0.5 & 0 & 0 & 1 \\ 0 & 0 & 0.8 & 0 \\ 0.7 & 0 & 1 & 0.4 \end{pmatrix}, \quad P_{\min} = \begin{pmatrix} 0 & 0.8 & 0.7 & 0.5 \\ 0.5 & 0 & 0 & 1 \\ 0 & 0 & 0.8 & 0 \\ 0.7 & 0 & 1 & 0.4 \end{pmatrix}.$$

图 2 是以 $P_{\max}$ 为转移矩阵的 4 个状态的 GPKS, 图 3 是以 $P_{\min}$ 为转移矩阵的 4 个状态的 GPKS.

为了研究 GPDP 上的模型检测问题, 下文引入调度 (schedulers) 来解决模型中的非确定性问题.

定义4 设 $M = (S, \mathrm{Act}, P, I, \mathrm{AP}, L)$ 是 GPDP, M 的调度定义为 $\mathrm{Adv} : S^{+} \to \mathrm{Act}$ 的函数, 对所有 $s_0 s_1 \cdots s_n \in S^{+}$, 使得 $\mathrm{Adv}(s_0 s_1 \cdots s_n) \in \mathrm{Act}(s_n)$.

对所有的 $i > 0$, 若 $\alpha_i = \mathrm{Adv}(s_0 s_1 \cdots s_{i-1})$, 则称路径 $\pi = s_0\alpha_1 s_1\alpha_2 s_2\alpha_3 s_3\alpha_4 s_4 \cdots$ 为 M 的 Adv 路径, 用 $\mathrm{Paths}_{\mathrm{Adv}}(s)$ 表示在调度 Adv 作用下, 从 s 出发的所有路径集, $\mathrm{Paths}_{\mathrm{Adv}}(M)$ 和 $\mathrm{Paths}^{\mathrm{fin}}_{\mathrm{Adv}}(M)$ 分别表示在调度 Adv 作用下, 从 M 中的所有状态出发的路径集合和有穷路径的集合.

与 GPKS 相比, GPDP 具有动作的非确定性和可能性分布. 因此, 通过调度可以诱导出一个 GPKS $M_{\mathrm{Adv}} = (S, P_{\mathrm{Adv}}, I, \mathrm{AP}, L_{\mathrm{Adv}})$, 对 $\sigma = s_0 s_1 \cdots s_n$, 有 $P_{\mathrm{Adv}}(\sigma, \sigma s_{n+1}) = P(s_n, \mathrm{Adv}(\sigma), s_{n+1})$, $L_{\mathrm{Adv}}(\sigma) = L(s_n)$. 对 GPDP M 的 Adv 路径 $L_{\mathrm{Adv}}(\sigma) = L(s_n)$, 在 GPKS M_{Adv} 有一条路径 $\pi_{\mathrm{Adv}} = \hat{\pi}_0\hat{\pi}_1\hat{\pi}_2 \cdots$ 与之对应, 其中 $\hat{\pi}_n = s_0 s_1 \cdots s_n$. 反之, 对 GPKS M_{Adv} 中一条路径, 对任意的 $I(s_0) > 0$ 的状态 s_0 有 $\hat{\pi}_0$=s_0. 对 $n > 0$, GPDP M 中状态 s_n 有 $\hat{\pi}_n = \hat{\pi}_{n-1} s_n$, 使得 $P(s_{n-1}, \mathrm{Adv}(\hat{\pi}_{n-1}), s_n) > 0$, 从而有 $s_0\mathrm{Adv}(\hat{\pi}_0)s_1\mathrm{Adv}(\hat{\pi}_1)s_2\mathrm{Adv}(\hat{\pi}_2)s_3\mathrm{Adv}(\hat{\pi}_3)s_4 \cdots$ 是 M 的 Adv 路径. 因此, GPDP M 中的路径与 GPKS

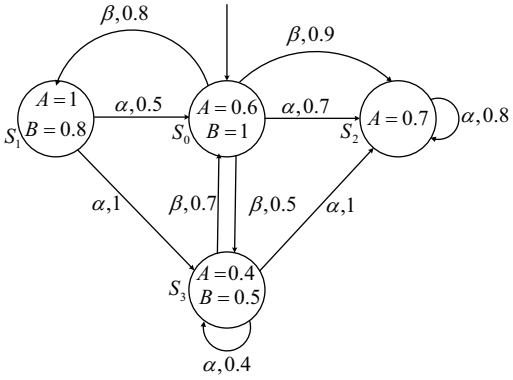

图 1 4 状态的 GPDP

Figure 1 Four states GPDP

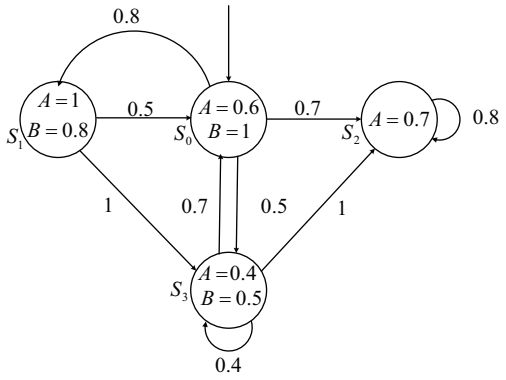

图 2 $P_{\max}$ 为转移矩阵的 GPKS

Figure 2 GPKS with respect to $P_{\max}$

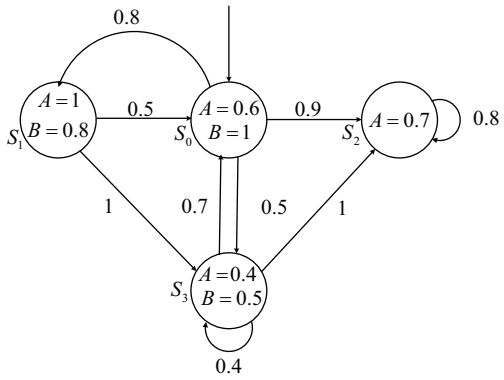

图 3 $P_{\min}$ 为转移矩阵的 GPKS

Figure 3 GPKS with respect to $P_{\min}$

M_{Adv} 中的路径是一一对应的, 本文把 GPKS M_{Adv} 的路径写成 GPDP M 的 Adv 路径.

下面给出 GPDP 在调度 Adv 下的柱集和广义可能性测度的定义.

定义5 给定一个 GPDP $M = (S, \mathrm{Act}, P, I, \mathrm{AP}, L)$, 设 $\hat{\pi} = s_0\alpha_0 s_1\alpha_1 s_2 \cdots s_{n-1}\alpha_{n-1}s_n \in \mathrm{Paths}^{\mathrm{fin}}_{\mathrm{Adv}}(M)$, 则有穷路径 $\hat{\pi}$ 的柱集定义为

$$\mathrm{Cyl}(\hat{\pi}) = \{\pi \in \mathrm{Paths}_{\mathrm{Adv}}(M) | \hat{\pi} \in \mathrm{Pref}(\pi)\}, \tag{1}$$

其中 $\mathrm{Pref}(\pi) = \{\pi' \in \mathrm{Paths}^{\mathrm{fin}}_{\mathrm{Adv}}(M) | \pi'$ 是 π 的有穷前缀$\}$.

定义6 设 $M = (S, \mathrm{Act}, P, I, \mathrm{AP}, L)$ 是有穷的 GPDP, 定义 $\mathrm{GPo} : \mathrm{Paths}_{\mathrm{Adv}}(M) \to [0,1]$ 如下:

$$\mathrm{GPo}(\pi) = I(s_0) \bigwedge \bigwedge_{i \geqslant 0} P_{\mathrm{Adv}}(s_i, \alpha_i, s_{i+1}), \tag{2}$$

其中 $\pi = s_0\alpha_0 s_1\alpha_1 s_2 \cdots \in \mathrm{Paths}_{\mathrm{Adv}}(M)$. 进一步, 对 $E \subseteq \mathrm{Paths}_{\mathrm{Adv}}(M)$, 定义 $\mathrm{GPo}(E) = \bigvee\{\mathrm{GPo}(\pi) | \pi \in E\}$, 从而得到函数

$$\mathrm{GPo} : 2^{\mathrm{Paths}_{\mathrm{Adv}}(M)} \to [0,1],$$

称为 $\Omega = 2^{\mathrm{Paths}_{\mathrm{Adv}}(M)}$ 上的广义可能性测度.

对于有穷的 GPDP $M=(S,\mathrm{Act},P,I,\mathrm{AP},L)$, $s\in S,\alpha_i\in\mathrm{Act},i\geqslant 0$, 定义 $r_{\mathrm{Adv}}:S\to[0,1]$ 为

$$r_{\mathrm{Adv}}(s)=\bigvee\left\{\bigwedge_{i\geqslant 0}P_{\mathrm{Adv}}(s_i,\alpha_i,s_{i+1})\,|s_1=s,s_i\in S,\alpha_i\in\mathrm{Act}\right\},\tag{3}$$

则 $r_{\mathrm{Adv}}(s)$ 表示从状态 s 出发的 Adv 路径最大可能性. 下面给出 r_{Adv} 的计算方法.

定理1 设 $M=(S,\mathrm{Act},P,I,\mathrm{AP},L)$ 是有穷的 GPDP, 对任意 $s\in S$, 有

$$r_{\mathrm{Adv}}(s)=\bigvee_{t\in S}\left(P^{+}_{\mathrm{Adv}}(s,t)\bigwedge P^{+}_{\mathrm{Adv}}(t,t)\right).\tag{4}$$

证明 (1) 对任意的 $s\in S$,

$$r_{\mathrm{Adv}}(s)=\bigvee\left\{\bigwedge_{i\geqslant 0}P_{\mathrm{Adv}}(s_i,\alpha_i,s_{i+1})\,|s_0=s,s_i\in S,\,\alpha_i\in\mathrm{Act}\right\}.$$

由于 S, Act 是有穷的, $\bigwedge$ 运算不会产生新的元素, 我们知道集合

$$\left\{\bigwedge_{i\geqslant 0}P_{\mathrm{Adv}}(s_i,\alpha_i,s_{i+1})\,|s_i\in S,\,\alpha_i\in\mathrm{Act}\right\}$$

是有穷的, 因此, 存在 Adv 路径 $\pi_{\mathrm{Adv}}=s\alpha_0s_1\alpha_1\cdots s_i\alpha_i$, 使得

$$r_{\mathrm{Adv}}(s)=\mathrm{GPo}(\pi_{\mathrm{Adv}})=P_{\mathrm{Adv}}\left(s,\alpha_0,s_1\right)\bigwedge P_{\mathrm{Adv}}\left(s_1,\alpha_i,s_2\right)\bigwedge\cdots.$$

GPDP M 中的路径与 GPKS M_{Adv} 中的路径是一一对应的, 对任意的状态 $s\in S$, $\mathrm{Act}(s)$ 是单点集. 因此,

$$P_{\mathrm{Adv}}\left(s,\alpha_0,s_1\right)\bigwedge P_{\mathrm{Adv}}\left(s_1,\alpha_i,s_2\right)\bigwedge\cdots=P_{\mathrm{Adv}}\left(s,s_1\right)\bigwedge P_{\mathrm{Adv}}\left(s_1,s_2\right)\bigwedge\cdots.$$

由于 S,Act 是有穷的, 则存在 $t\in S,i<j$, 使得 $s_i=s_j=t$, 从而得到

$$\begin{aligned}
&P_{\mathrm{Adv}}\left(s,s_1\right)\bigwedge P_{\mathrm{Adv}}\left(s_1,s_2\right)\bigwedge\cdots\\
&\quad=P_{\mathrm{Adv}}\left(s,s_1\right)\bigwedge\cdots\bigwedge P_{\mathrm{Adv}}\left(s_{i-1},t\right)\bigwedge\left(P_{\mathrm{Adv}}\left(t,s_{i+1}\right)\bigwedge\cdots\bigwedge P_{\mathrm{Adv}}\left(t,s_{j+1}\right)\right)\bigwedge\cdots\\
&\quad\leqslant P_{\mathrm{Adv}}\left(s,s_1\right)\bigwedge\cdots\bigwedge P_{\mathrm{Adv}}\left(s_{i-1},t\right)\bigwedge\left(P_{\mathrm{Adv}}\left(t,s_{i+1}\right)\bigwedge\cdots\bigwedge P_{\mathrm{Adv}}\left(t,s_{j+1}\right)\right)\\
&\quad\leqslant P^{+}_{\mathrm{Adv}}(s,t)\bigwedge P^{+}_{\mathrm{Adv}}(t,t).
\end{aligned}$$

因此,

$$r_{\mathrm{Adv}}\left(s\right)\leqslant\bigvee\left\{P^{+}_{\mathrm{Adv}}(s,t)\bigwedge P^{+}_{\mathrm{Adv}}(t,t)\,|t\in S\right\}.$$

(2) 对任意的 $t\in S$, 由 P^{+}_{Adv} 的定义知存在 $ss_1\cdots s_i=t\in S$ 和 $s_{i+1}\cdots s_i$, 使得

$$\begin{aligned}
P^{+}_{\mathrm{Adv}}\left(s,t\right)&=P_{\mathrm{Adv}}(s,s_1)\bigwedge\cdots\bigwedge P_{\mathrm{Adv}}\left(s_{i-1},t\right),\\
P^{+}_{\mathrm{Adv}}\left(t,t\right)&=P_{\mathrm{Adv}}(t,s_1)\bigwedge\cdots\bigwedge P_{\mathrm{Adv}}\left(s_j,t\right).
\end{aligned}$$

设 $\pi_{\mathrm{Adv}}=s_0\alpha_0s_1\alpha_1\cdots s_{i-1}\alpha_{i-1}t\alpha_i(s_{i+1}\cdots s_j\alpha_jt)^{\omega}$, 则

$$P^{+}_{\mathrm{Adv}}(s,t)\bigwedge P^{+}_{\mathrm{Adv}}(t,t)=P_{\mathrm{Adv}}(s,s_1)\bigwedge P_{\mathrm{Adv}}(s_1,s_2)\bigwedge\cdots.$$

由此得到

$$P^{+}_{\mathrm{Adv}}(s,t)\bigwedge P^{+}_{\mathrm{Adv}}(t,t)\leqslant r_{\mathrm{Adv}}(s),\quad \bigvee\left\{P^{+}_{\mathrm{Adv}}(s,t)\bigwedge P^{+}_{\mathrm{Adv}}(t,t)\,|t\in S\right\}\leqslant r_{\mathrm{Adv}}(s).$$

综合 (1) 和 (2) 得 $r_{\mathrm{Adv}}(s)=\bigvee\left\{P^{+}_{\mathrm{Adv}}(s,t)\bigwedge P^{+}_{\mathrm{Adv}}(t,t)\,|t\in S\right\}$.

用模糊矩阵计算形式为

$$r_{\mathrm{Adv}}=P^{+}_{\mathrm{Adv}}\circ D_{\mathrm{Adv}},\tag{5}$$

其中 $D_{\mathrm{Adv}}=\left(P^{+}_{\mathrm{Adv}}(t,t)\right)_{t\in S}$.

$r_{\mathrm{Adv}}(s)$ 的计算复杂性主要取决于计算可能性转移闭包时间, 根据文献 [25] 中计算转移闭包较优的算法, 我们知道其时间复杂度为 $O\left(n^2\log n\right)$, 其中 $n=|S|$.

定理2 设 $M=(S,\mathrm{Act},P,I,\mathrm{AP},L)$ 是有穷的 GPDP, 则 $\hat{\pi}=s_0\alpha_0s_1\alpha_1\cdots\alpha_{n-1}s_n\in\mathrm{Paths}^{\mathrm{fin}}_{\mathrm{Adv}}(M)$ 柱集的广义可能性测度为

$$\mathrm{GPo}\left(\mathrm{Cyl}(s_0\alpha_0s_1\alpha_1\cdots\alpha_{n-1}s_n)\right)=I(s_0)\bigwedge\bigwedge_{i=0}^{n-1}P_{\mathrm{Adv}}(s_i,\alpha_i,s_{i+1})\bigwedge r_{\mathrm{Adv}}(s_n),\tag{6}$$

其中 $\mathrm{GPo}\left(\mathrm{Cyl}(s_0)\right)=I(s_0)\bigwedge r_{\mathrm{Adv}}(s_0)$.

证明 注意到

$$\mathrm{Cyl}(s_0\alpha_0s_1\alpha_1\cdots\alpha_{n-1}s_n)=\bigcup\{\pi_{\mathrm{Adv}}\in\mathrm{Paths}_{\mathrm{Adv}}(M)\,|s_0\alpha_0s_1\alpha_1\cdots\alpha_{n-1}s_n\in\mathrm{Pref}(\pi_{\mathrm{Adv}})\}.$$

因此,

$$\begin{aligned}
&\mathrm{GPo}\left(\mathrm{Cyl}\left(s_0\alpha_0s_1\alpha_1\cdots\alpha_{n-1}s_n\right)\right)\\
&\quad=\bigvee\left\{\mathrm{GPo}\left(\pi_{\mathrm{Adv}}\right)|s_0\alpha_0s_1\alpha_1\cdots\alpha_{n-1}s_n\in\mathrm{Pref}\left(\pi_{\mathrm{Adv}}\right)\right\}\\
&\quad=\bigvee\left\{I\left(s_0\right)\bigwedge\bigwedge_{i\geqslant 0}P_{\mathrm{Adv}}\left(s_i,\alpha_i,s_{i+1}\right)|s_0\alpha_0s_1\alpha_1\cdots\alpha_{n-1}s_n\in\mathrm{Pref}\left(\pi_{\mathrm{Adv}}\right),\right.\\
&\qquad\left.s_{n+1},s_{n+2},\cdots\in S,\alpha_n\in\mathrm{Act}\right\}\\
&\quad=\left\{I\left(s_0\right)\bigwedge\bigwedge_{i=0}^{n-1}P_{\mathrm{Adv}}\left(s_i,\alpha_i,s_{i+1}\right)\right\}\bigwedge\bigvee\left\{\bigwedge_{j\geqslant n}P_{\mathrm{Adv}}\left(s_j,\alpha_j,s_{j+1}\right)|s_j\in S,\alpha_j\in\mathrm{Act}\right\}\\
&\quad=\left\{I\left(s_0\right)\bigwedge\bigwedge_{i=0}^{n-1}P_{\mathrm{Adv}}\left(s_i,\alpha_i,s_{i+1}\right)\right\}\bigwedge r_{\mathrm{Adv}}\left(s_n\right).
\end{aligned}$$

4 广义可能性计算树逻辑

本节引入文献 [17] 中的定量 GPoCTL 来描述有穷 GPDP 的性质. 不同的是, GPDP 下定义 GPoCTL 语义时不仅要考虑可能性分布, 且也要考虑调度. GPoCTL 由状态公式和路径公式构成. 下面给出 GPoCTL 的语法和在 GPDP 上的语义解释.

定义7 (GPoCTL 语法 [17]) 基于原子命题集合 AP 上的 GPoCTL 状态公式递归定义如下:

$$\Phi ::= \text{true}\,|a\,|\Phi_1 \bigwedge \Phi_2\,|\neg\Phi\,|\text{GPo}\ (\varphi)\,,$$

其中 $a \in \text{AP}$, φ 是 GPoCTL 的路径公式.

GPoCTL 路径公式递归定义如下:

$$\varphi ::= \bigcirc\Phi\,|\Phi_1 \bigcup \Phi_2|\Phi_1 \bigcup^{\leqslant n} \Phi_2\,|\Box\Phi,$$

其中 Φ, Φ_1, Φ_2 是状态公式, ϕ 是路径公式.

定义8 (GPoCTL 语义) 设 $M = (S, \text{Act}, P, I, \text{AP}, L)$ 是 GPDP, $a \in \text{AP}$, $s \in S, \Phi_1$, Φ_2 是 GPoCTL 的状态公式, ϕ 是 GPoCTL 的路径公式, 对状态公式 Φ, 其在 M 上的语义是 S 的模糊子集, 即 $\|\Phi\| : S \to [0,1]$. 归纳定义为对任意 $s \in S$,

$$\|\text{true}\|(s) = 1, \tag{7}$$

$$\|a\|(s) = L(s,a)\,, \tag{8}$$

$$\left\|\Phi_1 \bigwedge \Phi_2\right\|(s) = \|\Phi_1\|(s) \bigwedge \|\Phi_2\|(s)\,, \tag{9}$$

$$\|\neg\Phi\|(s) = 1 - \|\Phi\|(s)\,, \tag{10}$$

$$\|\text{GPo}\,(\varphi)\|(s) = \text{GPo}\,(s \models \varphi)\,. \tag{11}$$

令 $\pi = s_0\alpha_0 s_1\alpha_1 s_2 \cdots \in \text{Paths}_{\text{Adv}}(M), i \geqslant 0, \pi[i] = s_i$. 对路径公式 φ, 则 φ 在 M 上的语义是 $\|\varphi\| : \text{Paths}_{\text{Adv}}(M) \to [0,1]$. 归纳定义如下:

$$\|\bigcirc\Phi\|(\pi) = P_{\text{Adv}}(\pi[0], \alpha_0, \pi[1]) \bigwedge \|\Phi\|(\pi[1])\,,$$

$$\begin{aligned}\left\|\Phi_1 \bigcup \Phi_2\right\|(\pi) = \|\Phi_2\|(\pi[0]) \bigvee \bigvee_{j>0} \Big(&\|\Phi_1\|(\pi[0]) \bigwedge \bigwedge_{k<j} P_{\text{Adv}}(\pi[k-1], \alpha_{k-1}, \pi[k]) \\ &\bigwedge \|\Phi_1\|(\pi[k]) \bigwedge P_{\text{Adv}}(\pi[j-1], \alpha_{j-1}, \pi[j]) \bigwedge \|\Phi_2\|(\pi[j])\Big)\,,\end{aligned}$$

$$\begin{aligned}\left\|\Phi_1 \bigcup^{\leqslant n} \Phi_2\right\|(\pi) = \|\Phi_2\|(\pi[0]) \bigvee \bigvee_{0<j\leqslant n} \Big(&\|\Phi_1\|(\pi[0]) \bigwedge \bigwedge_{k<j} P_{\text{Adv}}(\pi[k-1], \alpha_{k-1}, \pi[k]) \\ &\bigwedge \|\Phi_1\|(\pi[k]) \bigwedge P_{\text{Adv}}(\pi[j-1], \alpha_{j-1}, \pi[j]) \bigwedge \|\Phi_2\|(\pi[j])\Big)\,,\end{aligned}$$

$$\|\Box\Phi\|(\pi) = \bigwedge_{i=0}^{\infty} \bigwedge_{j=0}^{i-1} P_{\text{Adv}}(\pi[j], \alpha_j, \pi[j+1]) \bigwedge \|\Phi\|(\pi[i])\,,$$

$$\|\Diamond\Phi\|(\pi) = \bigvee_{i=0}^{\infty} \bigwedge_{j=0}^{i-1} P_{\text{Adv}}(\pi[j], \alpha_j, \pi[j+1]) \bigwedge \|\Phi\|(\pi[i])\,.$$

$\text{GPo}\,(s \models \varphi)$ 表示从状态 s 出发, 在所有调度 Adv 下的所有满足公式 φ 路径的可能性. 定义如下:

$$\text{GPo}\,(s \models \varphi) = \bigvee_{\pi \in \text{Paths}_{\text{Adv}}(s)} (\text{GPo}\,(\pi) \bigwedge \|\varphi\|(\pi)). \tag{12}$$

5 广义可能性计算树模型检测

GPDP 上的 GPoCTL 模型检测问题描述为: 对于一个给定的 GPDPM, M 中的状态 s 和 GPoCTL 公式 Φ, 计算 $\|\Phi\|(s)$ 的值. 在所有调度 Adv 下, 可以在 $|\Phi|$ 步递归计算出 $\|\Phi\|(s)$ 的值, 这里 $|\Phi|$ 表示公式 Φ 的子公式数, 其递归定义如下:

如果 $\Phi \in \mathrm{AP} \bigcup \{\mathrm{true}\}$, 则 $|\Phi| = 1$.

$|\Phi_1 \bigwedge \Phi_2| = |\Phi_1| + |\Phi_2| + 1$.

$|\neg\Phi| = |\Phi| + 1$.

$|\mathrm{GPo}(\bigcirc\Phi)| = |\Box\mathrm{GPo}(\Phi)| = |\Phi| + 1$.

$|\mathrm{GPo}(|\Phi_1 \bigcup \Phi_2)| = \left|\mathrm{GPo}(|\Phi_1 \bigcup^{\leqslant n} \Phi_2)\right| = |\Phi_1| + |\Phi_2| + 1$.

对于式 $\Phi = a \in \mathrm{AP}$, $\Phi{=}\Phi_1 \bigwedge \Phi_2$ 和 $\Phi{=}\neg\Phi_2$, $\|\Phi\|(s)$ 可分别由式 (8)~(10) 计算出. 而对于式 $\Phi = \mathrm{GPo}(\varphi)$, $\|\Phi\|(s) = \mathrm{GPo}(s \models \varphi) = \bigvee_{\pi \in \mathrm{Paths}_{\mathrm{Adv}}(s)} (\mathrm{GPo}(\pi) \bigwedge \|\varphi\|(\pi))$, 理论上必须计算出所有调度 Adv 下的 $\|\Phi\|(s)$ 的值, 然后再算出状态 s 满足公式 Φ 的最大可能性和最小可能性. 在本节 GPoCTL 模型检测算法中, 我们给出了一种直接用模糊矩阵之间的运算来计算状态 s 满足公式 Φ 的最大可能性和最小可能性的算法. 用 $\|\Phi\|_{\max}(s)$ 和 $\|\Phi\|_{\min}(s)$ 分别表示状态 s 满足公式 Φ 的最大可能性和最小可能性. 下面分别给出路径公式 $\varphi = \bigcirc\Phi$, $\varphi = \Phi_1 \bigcup \Phi_2$, $\varphi = \Phi_1 \bigcup^{\leqslant n} \Phi_2$ 和 $\varphi = \Phi$ 分别对应的 $\|\Phi\|_{\max}(s)$ 和 $\|\Phi\|_{\min}(s)$ 的情况.

(1) 对于 $\varphi = \bigcirc\Phi$, 最大值 $\|\bigcirc\Phi\|_{\max}(s)$ 和最小值 $\|\bigcirc\Phi\|_{\min}(s)$ 计算过程分别如下:

$$
\begin{aligned}
&\|\mathrm{GPo}(\bigcirc\Phi)\|_{\max}(s)\\
&\quad = \mathrm{GPo}_{\max}(s \models \bigcirc\Phi)\\
&\quad = \bigvee_{\pi = s_0\alpha_0 s_1\alpha_1 s_2 \cdots \in \mathrm{Paths}_{\max}(s)} \mathrm{GPo}_{\max}(\pi) \bigwedge \|\bigcirc\Phi\|(\pi)\\
&\quad = \bigvee_{\pi = s_0\alpha_0 s_1\alpha_1 s_2 \cdots \in \mathrm{Paths}_{\max}(s)} \bigvee_{\alpha_0 \in \mathrm{Act}(s)} P(s,\alpha,s_1) \bigwedge \bigvee_{\alpha_1 \in \mathrm{Act}(s_1)} P(s_1,\alpha,s_2) \bigwedge \cdots \bigwedge\\
&\qquad \bigvee_{\alpha_0 \in \mathrm{Act}(s)} P(s,\alpha,s_1) \bigwedge \|\Phi\|(s_1)\\
&\quad = \bigvee_{\pi_{\max} = s_0\alpha_0 s_1\alpha_1 s_2 \cdots \in \mathrm{Paths}_{\max}(s)} P_{\max}(s,s_1) \bigwedge P_{\max}(s_1,s_2) \bigwedge \cdots \bigwedge P_{\max}(s,s_1) \bigwedge \|\Phi\|(s_1)\\
&\quad = \bigvee_{s_1 \in S} P_{\max}(s,s_1) \bigwedge \|\Phi\|(s_1) \bigwedge \bigvee_{s_2,s_3,\ldots \in S} \left(P_{\max}(s_1,s_2) \bigwedge P_{\max}(s_2,s_3) \bigwedge \cdots\right)\\
&\quad = \bigvee_{s_1 \in S} P_{\max}(s,s_1) \bigwedge \|\Phi\|(s_1) \bigwedge r_{\max}(s_1).
\end{aligned}
$$

对状态公式 Φ, D_Φ 表示 $|S| \times |S|$ 的模糊对角线矩阵, 当 $s = t$ 时, $D(s,t) = \|\Phi\|(s)$, 否则 $D(s,t) = 0$, $\mathrm{GPo}_{\max}(\bigcirc\Phi)$ 的矩阵计算形式为

$$\mathrm{GPo}_{\max}(s \models \bigcirc\Phi)_{s \in S} = P_{\max} \circ D_\Phi \circ r_{\max}, \tag{13}$$

其中模糊矩阵 $P_{\max}$ 表示在所有调度 Adv 下的 GPDP 对应的最大转移矩阵, r_{Adv} 表示以模糊矩阵 $P_{\max}$ 为转移矩阵的 GPKS 中状态 s 出发的最大可能性.

$\|\mathrm{GPo}(\bigcirc\Phi)\|_{\min}(s)$

$$
\begin{aligned}
&= \mathrm{GPo}_{\min}\left(s \mid= \bigcirc\Phi\right)\\
&= \bigvee_{\pi=s_0\alpha_0s_1\alpha_1s_2\cdots\in\mathrm{Paths}_{\min}(s)} \mathrm{GPo}_{\min}(\pi)\bigwedge\|\bigcirc\Phi\|(\pi)\\
&= \bigvee_{\pi=s_0\alpha_0s_1\alpha_1s_2\cdots\in\mathrm{Paths}_{\min}(s)} \bigwedge_{\alpha_0\in\mathrm{Act}(s)} P(s,\alpha,s_1)\bigwedge\bigwedge_{\alpha_1\in\mathrm{Act}(s_1)} P(s_1,\alpha,s_2)\bigwedge\cdots\bigwedge\\
&\quad \bigwedge_{\alpha_0\in\mathrm{Act}(s)} P(s,\alpha,s_1)\bigwedge\|\Phi\|(s_1)\\
&= \bigvee_{\pi=s_0\alpha_0s_1\alpha_1s_2\cdots\in\mathrm{Paths}_{\min}(s)} P_{\min}(s,s_1)\bigwedge P_{\min}(s_1,s_2)\bigwedge\cdots\bigwedge P_{\min}(s,s_1)\bigwedge\|\Phi\|(s_1)\\
&= \bigvee_{s_1\in S} P_{\min}(s,s_1)\bigwedge\|\Phi\|(s_1)\bigwedge\bigvee_{s_2,s_3,\ldots\in S}\left(P_{\min}(s_1,s_2)\bigwedge P_{\min}(s_2,s_3)\bigwedge\cdots\right)\\
&= \bigvee_{s_1\in S} P_{\min}(s,s_1)\bigwedge\|\Phi\|(s_1)\bigwedge r_{\min}(s_1).
\end{aligned}
$$

$\mathrm{GPo}_{\min}(\bigcirc\Phi)$ 的矩阵计算形式为

$$
\mathrm{GPo}_{\min}(s \mid= \bigcirc\Phi)_{s\in S} = P_{\min}\circ D_{\Phi}\circ r_{\min}, \tag{14}
$$

其中模糊矩阵 $P_{\min}$ 表示在所有调度 Adv 下的 GPDP 对应的最小转移矩阵, r_{Adv} 表示以模糊矩阵 $P_{\min}$ 为转移矩阵的 GPKS 中状态 s 出发的最大可能性.

(2) 对于 $\varphi=\Phi_1\bigcup^{\leqslant n}\Phi_2$, 最大值 $\left\|\mathrm{GPo}(\Phi_1\bigcup^{\leqslant n}\Phi_2)\right\|_{\max}(s)$ 和最小值 $\left\|\mathrm{GPo}(\Phi_1\bigcup^{\leqslant n}\Phi_2)\right\|_{\min}(s)$ 计算过程如下:

$$
\begin{aligned}
&\left\|\mathrm{GPo}(\Phi_1\bigcup^{\leqslant n}\Phi_2)\right\|_{\max}(s)\\
&= \mathrm{GPo}_{\max}\left(s \mid= \Phi_1\bigcup^{\leqslant n}\Phi_2\right)\\
&= \bigvee_{\pi=s_0\alpha_0s_1\alpha_1s_2\cdots\in\mathrm{Paths}_{\max}(s)} \bigvee_{\alpha_0\in\mathrm{Act}(s)} P(s,\alpha_0,s_1)\bigwedge\bigvee_{\alpha_1\in\mathrm{Act}(s_1)} P(s_1,\alpha_1,s_2)\bigwedge\cdots\bigwedge\|\Phi_2\|(s_1)\bigvee\\
&\quad \bigvee_{0<j\leqslant n}\left\{\|\Phi_1\|(s)\bigwedge\bigwedge_{k<j}\left(\bigvee_{\alpha_{k-1}\in\mathrm{Act}(s_{k-1})} P(s_{k-1},\alpha_{k-1},s_k)\bigwedge\|\Phi_1\|(s_k)\right)\bigwedge\right.\\
&\quad \left.\bigvee_{\alpha_{j-1}\in\mathrm{Act}(s_{j-1})} P(s_{j-1},\alpha_{j-1},s_j)\bigwedge\|\Phi_2\|(s_j)\right\}\\
&= \bigvee_{\pi=s_0\alpha_0s_1\alpha_1s_2\cdots\in\mathrm{Paths}_{\max}(s)} P_{\max}(s,s_1)\bigwedge P_{\max}(s_1,s_2)\bigwedge\cdots\bigwedge\|\Phi_2\|(s)\bigvee\\
&\quad \bigvee_{0<j\leqslant n}\left\{\|\Phi_1\|(s)\bigwedge\bigwedge_{k<j}\left(P_{\max}(s_{k-1},s_k)\bigwedge\|\Phi_1\|(s_k)\right)\bigwedge P_{\max}(s_{j-1},s_j)\bigwedge\|\Phi_2\|(s_j)\right\}\\
&= \|\Phi_2\|(s)\bigwedge r_{\max}(s)\bigvee\\
&\quad \bigvee_{0<j\leqslant n}\left\{\|\Phi_1\|(s)\bigwedge\bigwedge_{k<j} P_{\max}(s_{k-1},s_k)\bigwedge\|\Phi_1\|(s_k)\bigwedge P_{\max}(s_{j-1},s_j)\bigwedge\|\Phi_2\|(s_j)\bigwedge r_{\max}(s_j)\right\}
\end{aligned}
$$

$$= \left\{ \bigvee_{i=0}^{n} \left(D_{\Phi_1} \circ P_{\max}\right)^i \circ D_{\Phi_2} \circ r_{\max} \right\}(s).$$

对任意的 $n \geqslant |S|$, 有 $\bigvee_{i=0}^{n} \left(D_{\Phi_1} \circ P_{\max}\right)^i = \left(D_{\Phi_1} \circ P_{\max}\right)^*$, 则可得到 $\mathrm{GPo}_{\max}(\Phi_1 \bigcup^{\leqslant n} \Phi_2)$ 的矩阵计算形式为

$$\mathrm{GPo}_{\max}\left(s \models \Phi_1 \bigcup^{\leqslant n} \Phi_2\right)_{s \in S} = \left(D_{\Phi_1} \circ P_{\max}\right)^* \circ D_{\Phi_2} \circ r_{\max}. \tag{15}$$

$$\begin{aligned}
&\left\| \mathrm{GPo}(\Phi_1 \bigcup^{\leqslant n} \Phi_2) \right\|_{\min}(s) \\
&= \mathrm{GPo}_{\min}\left(s \models \Phi_1 \bigcup^{\leqslant n} \Phi_2\right) \\
&= \bigvee_{\pi = s_0\alpha_0 s_1 \alpha_1 s_2 \cdots \in \mathrm{Paths}_{\max}(s)} \bigwedge_{\alpha_0 \in \mathrm{Act}(s)} P(s, \alpha_0, s_1) \bigwedge \bigwedge_{\alpha_1 \in \mathrm{Act}(s_1)} P(s_1, \alpha_1, s_2) \bigwedge \cdots \bigwedge \|\Phi_2\|(s_1) \bigvee \\
&\quad \bigvee_{0 < j \leqslant n} \left\{ \|\Phi_1\|(s) \bigwedge \bigwedge_{k<j} \left(\bigwedge_{\alpha_{k-1} \in \mathrm{Act}(s_{k-1})} P(s_{k-1}, \alpha_{k-1}, s_k) \bigwedge \|\Phi_1\|(s_k) \right) \bigwedge \right. \\
&\quad \left. \bigwedge_{\alpha_{j-1} \in \mathrm{Act}(s_{j-1})} P(s_{j-1}, \alpha_{j-1}, s_j) \bigwedge \|\Phi_2\|(s_j) \right\} \\
&= \bigvee_{\pi = s_0\alpha_0 s_1 \alpha_1 s_2 \cdots \in \mathrm{Paths}_{\min}(s)} P_{\min}(s, s_1) \bigwedge P_{\min}(s_1, s_2) \bigwedge \cdots \bigwedge \|\Phi_2\|(s) \bigvee \\
&\quad \bigvee_{0 < j \leqslant n} \left\{ \|\Phi_1\|(s) \bigwedge \bigwedge_{k<j} \left(P_{\min}(s_{k-1}, s_k) \bigwedge \|\Phi_1\|(s_k) \right) \bigwedge P_{\min}(s_{j-1}, s_j) \bigwedge \|\Phi_2\|(s_j) \right\} \\
&= \|\Phi_2\|(s) \bigwedge r_{\min}(s) \bigvee \\
&\quad \bigvee_{0 < j \leqslant n} \left\{ \|\Phi_1\|(s) \bigwedge \bigwedge_{k<j} P_{\min}(s_{k-1}, s_k) \bigwedge \|\Phi_1\|(s_k) \bigwedge P_{\min}(s_{j-1}, s_j) \bigwedge \|\Phi_2\|(s_j) \bigwedge r_{\min}(s_j) \right\} \\
&= \left\{ \bigvee_{i=0}^{n} \left(D_{\Phi_1} \circ P_{\min}\right)^i \circ D_{\Phi_2} \circ r_{\min} \right\}(s).
\end{aligned}$$

对任意的 $n \geqslant |S|$, 有 $\bigvee_{i=0}^{n} \left(D_{\Phi_1} \circ P_{\min}\right)^i = \left(D_{\Phi_1} \circ P_{\min}\right)^*$, 则可得到 $\mathrm{GPo}_{\min}(\Phi_1 \bigcup^{\leqslant n} \Phi_2)$ 的矩阵计算形式为

$$\mathrm{GPo}_{\min}\left(s \models \Phi_1 \bigcup^{\leqslant n} \Phi_2\right)_{s \in S} = \left(D_{\Phi_1} \circ P_{\min}\right)^* \circ D_{\Phi_2} \circ r_{\min}. \tag{16}$$

(3) 对任意状态 s, 由 $\Phi_1 \bigcup \Phi_2 = \lim_{n \to \infty} \Phi_1 \bigcup^{\leqslant n} \Phi_2$ 得 $\mathrm{GPo}(s \models \Phi_1 \bigcup \Phi_2) = \lim_{n \to \infty} \|\mathrm{GPo}(\Phi_1 \bigcup^{\leqslant n} \Phi_2)\|(s)$, 从而得到最大值 $\|\mathrm{GPo}_{\max}(\Phi_1 \bigcup \Phi_2)\|(s)$ 和最小值 $\|\mathrm{GPo}_{\min}(\Phi_1 \bigcup \Phi_2)\|(s)$ 的矩阵计算公式为

$$\left(\left\| \mathrm{GPo}_{\max}(\Phi_1 \bigcup \Phi_2) \right\|(s) \right)_{s \in S} = \left(D_{\Phi_1} \circ P_{\max}\right)^* \circ D_{\Phi_2} \circ r_{\max}, \tag{17}$$

$$\left(\left\| \mathrm{GPo}_{\min}(\Phi_1 \bigcup \Phi_2) \right\|(s) \right)_{s \in S} = \left(D_{\Phi_1} \circ P_{\min}\right)^* \circ D_{\Phi_2} \circ r_{\min}. \tag{18}$$

(4) 对于 $\varphi = \Box\Phi$, 最大值 $\mathrm{GPo}_{\max}(s \models \Box\, \Phi)$ 和最小值 $\mathrm{GPo}_{\min}(s \models \Box\, \Phi)$ 计算过程如下:

$$\mathrm{GPo}_{\max}(s \models \Box\, \Phi)$$

$$
\begin{aligned}
&= \bigvee_{\pi=s_0\alpha_0s_1\alpha_1s_2\cdots\in \mathrm{Paths}_{\max}(s)} \mathrm{GPo}(\pi) \bigwedge \|\Box\Phi\|(\pi) \\
&= \bigvee_{\pi=s_0\alpha_0s_1\alpha_1s_2\cdots\in \mathrm{Paths}_{\max}(s)} \bigwedge_{j=0}^{\infty} \left\{ \bigvee_{\alpha_j\in \mathrm{Act}(s_j)} P(s_j,\alpha_j,s_{j+1}) \right\} \bigwedge_{j=0}^{\infty} \|\Phi\|(s_j) \\
&= \bigvee_{\pi=s_0\alpha_0s_1\alpha_1s_2\cdots\in \mathrm{Paths}_{\max}(s)} \bigwedge_{j=0}^{\infty} P_{\max}(s_j,s_{j+1}) \bigwedge_{j=0}^{\infty} \|\Phi\|(s_j),
\end{aligned}
$$

$$
\begin{aligned}
&\mathrm{GPo}_{\min}(s \models \Box\,\Phi) \\
&= \bigvee_{\pi=s_0\alpha_0s_1\alpha_1s_2\cdots\in \mathrm{Paths}_{\min}(s)} \mathrm{GPo}(\pi) \bigwedge \|\Box\Phi\|(\pi) \\
&= \bigvee_{\pi=s_0\alpha_0s_1\alpha_1s_2\cdots\in \mathrm{Paths}_{\min}(s)} \bigwedge_{j=0}^{\infty} \left\{ \bigwedge_{\alpha_j\in \mathrm{Act}(s_j)} P(s_j,\alpha_j,s_{j+1}) \right\} \bigwedge_{j=0}^{\infty} \|\Phi\|(s_j) \\
&= \bigvee_{\pi=s_0\alpha_0s_1\alpha_1s_2\cdots\in \mathrm{Paths}_{\min}(s)} \bigwedge_{j=0}^{\infty} P_{\min}(s_j,s_{j+1}) \bigwedge_{j=0}^{\infty} \|\Phi\|(s_j).
\end{aligned}
$$

与前面的公式不同, 公式 $\mathrm{GPo}(\Box\Phi)$ 不能用模糊矩阵的形式来计算, 由模糊矩阵 $P_{\max}$ 和模糊矩阵 $P_{\min}$ 可以构造两个 GPKS, 故采用不动点算法 (算法 1) 来计算.

算法 1 最大不动点算法

Require: 函数 $f(Z)=\|\Phi\|\bigwedge\|\mathrm{GPo}(\bigcirc Z)\|$, 其中 $\|\mathrm{GPo}(\bigcirc Z)\|=P_{\mathrm{Adv}}\circ D_Z\circ r_{\mathrm{Adv}}$;
Ensure: f 的最大不动点;

```
Procedure Fixpoint (f, x)
    x' ← f(x)
    While x ≠ x' do
        x ← x'
        x' ← f(x)
    End while
    Return x
End procedure
```

引理1 ([17]) 对有穷的 GPKS $M_{\mathrm{Adv}}=(S,P_{\mathrm{Adv}},I,\mathrm{AP},L)$ 和任意的 GPoCTL 公式 Φ, 定义函数 $f(Z)=\|\Phi\|\bigwedge\|\mathrm{GPo}(\bigcirc Z)\|$, 其中 $\|\mathrm{GPo}(\bigcirc Z)\|=P_{\mathrm{Adv}}\circ D_Z\circ r_{\mathrm{Adv}}$, 函数 f 有最大不动点且为 $\|\mathrm{GPo}(\Phi)\|$.

设 $M=(S,P,I,\mathrm{AP},L)$ 是有穷的 GPDP, 对任意的 $s\in S$, Adv 是 $\|\Phi\|_{\max}(s)$ 和 $\|\Phi\|_{\min}(s)$ 对应的所有调度. 当计算 $\|\Phi\|_{\max}(s)$ 时 $P_{\mathrm{Adv}}=P_{\max}$, $r_{\mathrm{Adv}}=r_{\max}$, 当计算 $\|\Phi\|_{\min}(s)$ 时 $P_{\mathrm{Adv}}=P_{\min}$, $r_{\mathrm{Adv}}=r_{\min}$. $r_{\mathrm{Adv}}=P^{+}_{\mathrm{Adv}}\circ D_{\mathrm{Adv}}$, $D_{\mathrm{Adv}}=(P^{+}_{\mathrm{Adv}}(s,s))_{s\in S}$, $P^{+}_{\mathrm{Adv}}=P^{1}_{\mathrm{Adv}}\bigvee P^{2}_{\mathrm{Adv}}\bigvee\cdots P^{|S|}_{\mathrm{Adv}}$, $P^{*}_{\mathrm{Adv}}=P^{0}_{\mathrm{Adv}}\bigvee P^{+}_{\mathrm{Adv}}$, P^{0}_{Adv} 是 $|S|\times|S|$ 的单位矩阵, $D_\Phi=\mathrm{diag}(\|\Phi\|(s))_{s\in S}$, 根据式 (7)∼(18), 给出具体的 GPoCTL 模型检测算法 (算法 2).

在 GPoCTL 模型检测算法中, 公式 $\Phi=a\in\mathrm{AP}$, $\Phi{=}\Phi_1\bigwedge\Phi_2$ 和 $\Phi{=}\neg\Phi_2$, 计算 $\|\Phi\|(s)$ 的时间只与 GPDP M 的大小和公式 Φ 的长度有关. 而计算公式 $\Phi=\mathrm{GPo}(\varphi)$ 的时间主要取决于 P_{Adv} 的转移矩阵闭包 P^{+}_{Adv} 时间和计算不动点的时间. 我们采用文献 [25] 的算法来计算 P^{+}_{Adv}, 其时间复杂度为 $O\left(n^2\log n\right)$, 其中 $n=|S|$. 对于不动点 $f(Z_\Phi)=\|\Phi\|\bigwedge\mathrm{GPo}(\bigcirc Z)$ 计算时间, 正好是从状态 s 出发

算法 2 GPoCTL 模型检测算法

Require: GPDP M 和 GPoCTL 公式 Φ;
Ensure: 对 M 中任意的状态 s, $\|\Phi\|_{\mathrm{Adv}}(s)$ 的值
Procedure procedure GPoCTLCheck(Φ)
Case Φ
true return$(1)_{s\in S}$
$a \in \mathrm{AP}$ return $(\|a\|(s))_{s\in S}$
$\neg\Phi$ return $(1-\|\Phi\|(s))_{s\in S}$
$\Phi_1 \bigwedge \Phi_2$ return$(\|\Phi_1\|(s) \bigwedge \|\Phi_2\|(s))_{s\in S}$
$\mathrm{GPo}(\mathrm{O}\Phi)$ return $P_{\mathrm{Adv}} \circ D_{\Phi} \circ r_{\mathrm{Adv}}$
$\mathrm{GPo}\left(\Phi_1 \bigcup^{\leqslant n} \Phi_2\right)$ return $\bigvee_{i=0}^{n} \left(D_{\Phi_1} \circ P_{\mathrm{Adv}}\right)^i \circ D_{\Phi_2} \circ r_{\mathrm{Adv}}$
$\mathrm{GPo}(\Phi_1 \bigcup \Phi_2)$ return $\left(D_{\Phi_1} \circ P_{\mathrm{Adv}}\right)^* \circ D_{\Phi_2} \circ r_{\mathrm{Adv}}$
$\mathrm{GPo}(\Diamond\Phi)$ return $(P_{\mathrm{Adv}})^* \circ D_{\phi_2} \circ r_{\mathrm{Adv}}$
$\mathrm{GPo}(\Box\Phi)$ return Fixpoint $\left((1)_{s\in S}, f_{\Phi}\right)$
End case
End procedure

长度为 n 的所有有穷路径 $\hat{\pi}$ 满足公式 Φ 的最小上界. 由于状态空间 S 有穷, 则对于任意长度大于 $|S|+1$ 的路径 $\hat{\pi}'$, 其值大于 $\hat{\pi}$ 满足公式 Φ 的值. 由此得到不动点迭代计算 $|S|+1$ 次后收敛. 又因为计算不动点只是模糊矩阵的合成运算, 一个迭代计算时间为 $O(|S|^2)$, 从而得计算不动点算法时间复杂度为 $O(|S|^3)$. 综上所述, 我们通过定理 3 给出 GPoCTL 模型检测算法的时间复杂度.

定理3 (GPoCTL 模型检测算法的时间复杂度) 给定一个有穷的 GPDP $M=(S,\mathrm{Act},P,I,\mathrm{AP},L)$ 和一个 GPoCTL 公式 Φ, $M \models \Phi$ 的时间复杂度为 $O(\mathrm{size}(M)\cdot \mathrm{poly}(S)\cdot|\Phi|)$, 其中 $\mathrm{size}(M)$ 是模型的大小, $\mathrm{poly}(n)$ 是 n 的多项式函数, $|\Phi|$ 是公式的长度.

6 实例应用

我们采用类似于文献 [26~30] 中病人治疗的专家系统来说明 GPoCTL 模型检测技术的应用. 假设存在一种新的细菌感染疾病, 医生对此类疾病没有足够的理论知识来制定治疗方案, 只能凭借自己的经验来确定治疗方案. 我们用 GPDP $M=(S,\mathrm{Act},P,I,\mathrm{AP},L)$ 对病人治疗过程进行建模.

为方便起见, 假设医生将病人的健康状况分为 “差”、“一般” 和 “好” 3 种状态, 分别用 S_0,S_1,S_2 表示, 即 $S=\{S_0,S_1,S_2\}$. 为了对状态进行标记, 设 $\mathrm{AP}=\{P,G,E\}$, 其中 P, G 和 E 分别表示某种状态时病人身体健康状况为 “差”、“一般” 和 “好”. 针对病人的这 3 种健康状况, 不同的医生理解不同. 因此, 我们给他们赋一个模糊值, 表示身体健康状况的程度. 例如 $L(S_2,E)=0.9$ 表示在状态 S_2 上健康状况为 “好” 的程度是 0.9. 设 $\mathrm{Act}=\{\alpha,\beta,\gamma\}$, 表示医生采用 α,β,γ 3 种不同的治疗方案对病人进行治疗. 用 $P(S_0,\alpha,S_1)=0.3$ 表示医生采用 α 方案对病人治疗后, 病人的健康状况从状态 S_0 到状态 S_1 的可能性为 0.3. 为了使图形简洁, 图中的每条边对应着 α,β,γ 的 3 个可能性转移, 具体的 GPDP M 如图 4 所示.

我们得出初始分布 I, α,β,γ 对应的转移矩阵 $P_\alpha,P_\beta,P_\gamma$. P,G,E 在状态 S_0,S_1,S_2 对应的真值矩阵 P_P, P_G,P_E 分别为

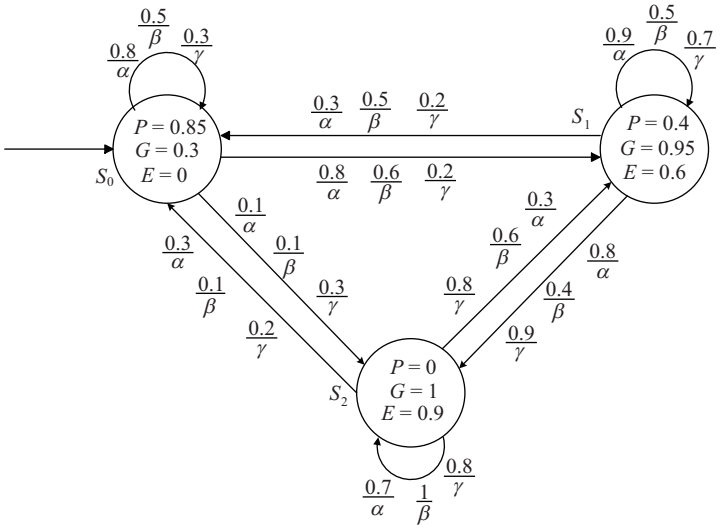

图 4　病人治疗过程 MPDP 模型 M

Figure 4　Treatment process of the patients modeled by MPDP M

$$I=\begin{pmatrix}1\\0\\0\end{pmatrix},\quad P_\alpha=\begin{pmatrix}0.8&0.8&0.1\\0.3&0.9&0.8\\0.3&0.7&0.3\end{pmatrix},\quad P_\beta=\begin{pmatrix}0.5&0.6&0.1\\0.5&0.5&0.4\\0.1&0.8&1\end{pmatrix},\quad P_\gamma=\begin{pmatrix}0.3&0.2&0.3\\0.2&0.7&0.9\\0.2&0.8&0.8\end{pmatrix},$$

$$P_P=\begin{pmatrix}0.85&0&0\\0&0.4&0\\0&0&0\end{pmatrix},\quad P_G=\begin{pmatrix}0.3&0&0\\0&0.95&0\\0&0&1\end{pmatrix},\quad P_E=\begin{pmatrix}0&0&0\\0&0.6&0\\0&0&0.9\end{pmatrix}.$$

根据上面的矩阵, 计算出矩阵 $P_{\max}, P_{\min}, P^*_{\max}, P^*_{\min}$ 为

$$P_{\max}=\begin{pmatrix}0.8&0.8&0.3\\0.5&0.9&0.9\\0.3&0.8&1\end{pmatrix},\quad P_{\min}=\begin{pmatrix}0.3&0.2&0.1\\0.2&0.5&0.4\\0.1&0.3&0.7\end{pmatrix},\quad P^*_{\max}=\begin{pmatrix}0.8&0.8&0.8\\0.5&0.9&0.9\\0.5&0.8&1\end{pmatrix},\quad P^*_{\min}=\begin{pmatrix}0.3&0.2&0.1\\0.2&0.5&0.4\\0.2&0.3&0.7\end{pmatrix}.$$

根据算法 2, 计算出表 1 中状态 S_0, S_1, S_2 满足公式 Φ 的可能性. 对表 1 中的结果说明:

(1) $\|\mathrm{GPo}(\bigcirc E)\|_{\max}(S_0)=0.6$, $\|\mathrm{GPo}(\bigcirc E)\|_{\min}(S_0)=0.2$ 说明在状态 S_0, 医生采用 3 种治疗方案治疗一次后病人健康状况为 “好” 的最大可能性为 0.6, 最小可能性为 0.2.

(2) $\|\mathrm{GPo}(P\bigcup E)\|_{\max}(S_0)=0.6$, $\|\mathrm{GPo}(P\bigcup E)\|_{\min}(S_0)=0.6$ 说明在状态 S_0 病人的健康状况为 “差”, 医生采用 3 种治疗方案经过多次治疗后, 病人的健康状况为 “好” 的最大可能性为 0.6, 最小可能性为 0.6.

(3) $\|\mathrm{GPo}(\Diamond E)\|_{\max}(S_0)=0.8$, $\|\mathrm{GPo}(\Diamond E)\|_{\min}(S_0)=0.6$ 说明从状态 S_0 开始, 医生采用 3 种治疗方案经过多次治疗后, 病人的健康状况最终为 “好” 的最大可能性为 0.8, 最小可能性为 0.6.

(4) $\|\mathrm{GPo}(\Box E)\|_{\max}(S_0)=0$, $\|\mathrm{GPo}(\Box E)\|_{\min}(S_0)=0$ 说明从状态 S_0 开始, 医生采用 3 种治疗方案开始治疗, 病人的健康状况总是为 “好” 的最大可能性为 0, 最小可能性为 0.

表 1 $s \models \Phi$ 的可能性

Table 1 The possibility of $s \models \Phi$

Number	FormulaΦ	$\|\Phi\|_{\max}(s)$	$\|\Phi\|_{\min}(s)$
1	GPo($\bigcirc E$)	$\{0.6, 0.9, 0.9\}$	$\{0.2, 0.5, 0.7\}$
2	GPo($P \bigcup E$)	$\{0.6, 0.6, 0.9\}$	$\{0.6, 0.5, 0.7\}$
3	GPo($\Diamond E$)	$\{0.8, 0.9, 0.9\}$	$\{0.6, 0.5, 0.7\}$
4	GPo($\Box E$)	$\{0, 0.6, 0.9\}$	$\{0, 0.4, 0.7\}$

7 结论

为了解决非确定性系统的正确性验证问题, 本文研究了基于决策过程的 GPoCTL 模型检测问题. 首先给出了描述非确定性系统行为的 GPDP 的定义, 并在 GPDP 下定义了 GPoCTL 的语义. 然后提出了 GPoCTL 的模型检测算法. 进一步通过实例说明了 GPoCTL 的模型检测算法的应用. 本文只研究了定量的 GPoCTL 模型检测算法, 未来将继续研究 GPoCTL 的表达能力, 同时研究在 GPDP 下 LTL 模型检测问题.

致谢 感谢潘海玉博士与本文作者进行了有益讨论并提出的宝贵建议.

参考文献

1 Baier C, Katoen J P. Principles of Model Checking. Cambridge: MIT Press, 2008
2 Edmund M, Grumberg O, Peled D. Model Checking. Cambridge: MIT Press, 1999
3 Zheng Y, Wang L. Consensus of switched multi-agent systems. IEEE Trans Circ Syst II, 2016, 63: 314–318
4 Zheng Y, Wang L. A novel group consensus protocol for heterogeneous multi-agent systems. Int J Contr, 2015, 106: 1–13
5 Li T, Zhang J F. Consensus conditions of multi-agent systems with time-varying topologies and stochastic communication noises. IEEE Trans Automat Contr, 2010, 55: 2043–2057
6 Li T, Fu M, Xie L, et al. Distributed consensus with limited communication data rate. IEEE Trans Automat Contr, 2011, 56: 279–292
7 Baier C, Kwiatkowska M. Model checking for a probabilistic branching time logic with fairness. Distrib Comput, 1998, 11: 125–155
8 Hart S, Sharir M. Termination of probabilistic concurrent programs. ACM Trans Prog Lang Syst, 1983, 5: 356–380
9 Sultan K, Bentahar J, Wei W, et al. Modeling and verifying probabilistic multi-agent systems using knowledge and social commitments. Expert Syst Appl, 2014, 41: 6291–6304
10 Sultan K, Bentahar J, EI-Menshawy M. Model checking probabilistic social commitments for intelligent agent communication. Appl Softw Comput, 2014, 22: 397–409
11 Chechik M, Devereux B, Easterbrook S, et al. Multi-valued symbolic model-checking. ACM Trans Softw Eng Method, 2003, 12: 371–408
12 Chechik M, Gurfinkel A, Devereux B, et al. Data structures for symbolic multi-valued model-checking. Formal Methods Syst Des, 2006, 29: 295–344
13 Pan H Y, Li Y M, Cao Y Z, et al. Model checking fuzzy computation tree logic. Fuzzy Sets Syst, 2015, 262: 60–77
14 Pan H Y, Li Y M, Cao Y Z, et al. Model checking computation tree logic over finite lattices. Theor Comput Sci, 2016, 612: 45–62
15 Li Y M, Li L J. Model checking of linear-time properties based on possibility measure. IEEE Trans Fuzzy Syst, 2013, 21: 842–854
16 Li Y M, Li Y L, Ma Z Y. Computation tree logic model checking based on possibility measures. Fuzzy Sets Syst, 2015, 262: 44–59

17 Li Y M, Ma Z Y. Quantitative computation tree logic model checking based on generalized possibility measures. IEEE Trans Fuzzy Syst, 2015, 23: 2034–2047
18 Drakopoulos A. Probabilities, possibilities, and fuzzy sets. Fuzzy Sets Syst, 1995, 75: 1–15
19 Dubois D. Possibility theory and statistical reasoning. Comput Stat Data Anal, 2006, 51: 47–69
20 Dubois D, Prade H. Possibility Theory. NewYork: Plenum, 1988
21 Dubois D, Prade H. Possibility theory, probability theory and multiple-valued logics: a clarification. Ann Math Artif Intell, 2001, 32: 35–66
22 Grabisch M, Murofushi T, Sugeno M. Fuzzy Measures and Integrals. Heidelberg: Physica-Verlag, 2000
23 Zadeh L A. Fuzzy sets as a basis for a theory of possibility. Fuzzy Sets Syst, 1978, 1: 3–28
24 Li Y M. Analysis of Fuzzy Systems. Beijing: Science Press, 2005 [李永明. 模糊系统分析. 北京: 科学出版社, 2005]
25 Garmendia L, González del Campo R, López V, et al. An algorithm to compute the transitive closure, a transitive approximation and a transitive opening of a fuzzy proximity. Mathware Soft Comput, 2009, 16: 175–191
26 Lin F, Ying H. Modeling and control of fuzzy discrete event systems. IEEE Trans Syst Man Cybernetics Part B, 2002, 32: 408–415
27 Qiu D. Supervisory control of fuzzy discrete event systems: a formal approach. IEEE Trans Syst Man Cybern Part B, 2005, 35: 72–88
28 Cao Y, Ying M. Observability and decentralized control of fuzzy discrete-event systems. IEEE Trans Fuzzy Syst, 2006, 14: 202–216
29 Liu F C, Qiu D W. Diagnosability of fuzzy discrete event systems: a fuzzy approach. IEEE Trans Fuzzy Syst, 2009, 17: 372–384
30 Xing H Y, Zhang Q S, Huang K S. Analysis and control of fuzzy discrete event systems using bisimulation equivalence. Theor Comput Sci, 2012, 456: 100–111

Model checking generalized possibilistic computation tree logic based on decision processes

Zhanyou MA[1,2] & Yongming LI[1*]

1 *College of Computer Science, Shaanxi Normal University, Xi'an* 710119*, China*;
2 *College of Computer Science and Engineering, Beifang University of Nationalities, Yinchuan* 750021*, China*
*E-mail: liyongm@snnu.edu.cn

Abstract We study model-checking generalized possibilistic computation tree logic (GPoCTL) and its application in system verification, in particular, nondeterministic systems. Firstly, we introduce generalized possibilistic decision-making processes (GPDP) as system models and GPoCTL formulae under GPDP to describe the properties of the system. Then, we provide a model-checking algorithm for GPoCTL. The main advantage of the algorithm is that it can use scheduling in the decision making processes to convert the model-checking problem into operations of the fuzzy matrix or fixed point of fuzzy matrix functions, which need polynomial time. Finally, an example is given to illustrate the application of the model-checking generalized possibilistic computation tree logic in verification of a nondeterministic system.

Keywords nondeterministic system, generalized possibilistic decision process, scheduler, generalized possibilistic computation tree logic, model checking

Zhanyou MA was born in 1979. He received the M.S. degree in computer science from Beifang University of Nationalities, Yinchuan, China, in 2007. Since 2011, he has been working toward the Ph.D. degree in computer science at Shaanxi Normal University, Xi'an, China. He is also an associate professor of computer science at the Beifang University of Nationalities, Yinchuan, China. His research interests include multi-valued model checking and intelligent control.

Yongming LI was born in 1966. He received the Ph.D. degree in mathematics from Sichuan University, Chengdu, China, in 1996. He is currently with the Shaanxi Normal University, Xi'an, China, as a professor of mathematics and computer science. His research interests include model checking, fuzzy control theory, fuzzy automata theory, spatial reasoning, quantum logic and quantum computation, and topology over lattices.