

带数据约束的概率实时系统的验证

张春燕¹ 孙 俊²

(无锡科技职业学院 无锡 214000)¹ (江南大学 无锡 214000)²

摘 要 带数据约束的概率实时系统是指一种既带有概率时间约束又带有数据变量约束的计算系统。目前将离散数据约束和连续时间约束统一在一个概率模型中的规范及验证研究较少。提出了一种既带有连续数据约束又带有离散数据约束的规范——基于连续时间的概率 ZIA 规范,并给出了它的时序逻辑。对于 CTL 和 PCTL 而言,尽管这些逻辑很强大,但是只能反映时序性质,因此提出一个新的形式化语言 CTML 来表达度量性质查询,同时保留表达时序性质的能力并给出概率 ZIA 规范的验证算法。

关键词 数据约束,概率实时系统,连续时间,ZIA

中图法分类号 TP316.2 **文献标识码** A

Validation for Probability Real-time System with Data Constraints

ZHANG Chun-yan¹ SUN Jun²

(Wuxi Professional College of Science and Technology, Wuxi 214000, China)¹

(Jiangnan University, Wuxi 214000, China)²

Abstract A probabilistic real-time system with data constraints is a computing system with both probabilistic time constraints and data constraints. At present, there exist a few studies of the specification and verification that uniform discrete data constraints and sequence time constraint work in the probability model. In this paper, we proposed a specification based on sequence time probability ZIA, which has both sequence data anstraints and discrete data constraints, and gave its temporal logic. For CTL and PCTL, although the logic is very powerful, it can only reflect the temporal properties. This paper proposed a new formal language CTML to express the metric properties of query which retains the ability to express temporal properties and gave the validation algorithm of probability of ZIA specification.

Keywords Data constraints, Probability of real-time system, Continuous time, ZIA

1 引言

带数据约束的概率实时系统^[1]指一种既带有概率时间约束又带有数据变量约束的计算系统。飞行控制、核反应堆控制以及铁路调度控制等计算机控制系统都属于带数据约束的概率实时系统。这些系统中许多动作的完成都与时间相关,即要满足一定的时间限制,如某个动作要在一秒钟内完成;同时在这些系统中数据变量之间也有一定的约束关系,如飞机的飞行速度可能跟气压、高度等之间有一定的约束关系。对这类系统而言,确保其正确性和可靠性是至关重要的。

模型检测是一种自动化的、用于并发系统性质验证的算法技术^[2-3],但它只能对有穷状态系统进行性质验证。其基本思想是用状态迁移系统 S 表示系统的行为,用时序逻辑公式 F 来描述系统的性质。这样,“系统是否满足所期望的性质”就转化为数学问题“状态迁移系统 S 是否是公式 F 的一个模型”^[4]。

为了对带数据约束的概率实时系统进行研究,文献^[5-6]

提出了 ZIA 和 HZIA 规范并给出了 ZIA 和 HZIA 上的精化关系,但是并没有给出对应的时序逻辑以及模型检测算法。文献^[7]提出了一种基于离散时间的 ZIA^[8]规范,但是基于离散时间的模型^[9]较适用于同步系统,如果要利用离散时间来模拟异步系统,就必须选择一些固定的时间量子来离散化时间,使得系统中任意两个事件之间的延迟时间都是该时间量子的倍数,这点很难做到,并可能会限制系统模拟的准确性。因此,本文提出了一种基于连续时间的概率 ZIA 规范。

2 基于连续时间的概率 ZIA(CT-PZIA)

ZIA 可以同时刻画系统的行为和状态,却没有刻画概率实时方面的能力;而概率时间自动机是针对概率实时系统提出的一种模型。因此,将 ZIA 与概率时间自动机相结合,提出一种针对概率实时系统且能够同时刻画行为和状态的规范 CT-PZIA。

2.1 CT-PZIA 的定义

定义 1 接口自动机是一个六元组 $P1 = \langle V_P, V_P^i, A_P^I, \dots \rangle$,

本文受国家自然科学基金(61300149),江苏省教育厅高校哲学社会科学研究指导项目(2016SJD880064),江苏高校品牌专业建设工程资助项目资助。

张春燕 讲师,主要研究方向为人工智能算法与模式识别、嵌入式计算机控制系统等, E-mail: zcy0006@163.com; 孙 俊 教授,博士生导师,主要研究方向为人工智能、机器学习等。

A_p^O, A_p^H, TP , 由下列元素组成:

(1) V_p 是状态的集合。

(2) $V_p^i \subseteq V_p$ 是初始状态的集合。如果 $V_p^i = \emptyset$, 那么 P 就为空。

(3) A_p^I, A_p^O 和 A_p^H 分别是不相交的输入动作集合、输出动作集合和内部动作集合。记所有动作集合 $A_p = A_p^I \cup A_p^O \cup A_p^H$ 。

(4) TP 是状态之间转换关系的集合, 如 $T_p \subseteq V_p \times A_p \times V_p$ 。如果接口自动机 $P1$ 只有内部动作, 即 $A_p^I = A_p^O = \emptyset$, 那么 P 是封闭的, 否则说 $P1$ 是开放的。

定义 2 (子分布函数, 概率分布函数) 一个子分布函数是在一个可数集上的函数: $\mu: S \rightarrow [0, 1]$ 使得 $\sum_{s \in S} \mu(s) \leq 1$ 。当 $\sum_{s \in S} \mu(s) = 1$ 时, 称为简单的概率分布函数。令 $SDistr(S)$ 和 $Distr(S)$ 分别表示所有的分布函数和简单分布函数集。

定义 3 一个基于连续时间的概率 ZIA (CT-PZIA), P :

$\langle S_p, S_p^i, A_p^I, A_p^O, A_p^H, V_p^i, V_p^O, V_p^H, F_p^V, F_p^A, X, I, prob, T_p \rangle$ 。

其中: (1) S_p 是状态的集合。(2) $S_p^i \subseteq S_p$ 是初始状态的集合, 如果 $S_p^i = \emptyset$, 那么 P 就为空。(3) A_p^I, A_p^O 和 A_p^H 分别是不相交的输入动作集合、输出动作集合和内部动作集合。记所有动作集合 $A_p = A_p^I \cup A_p^O \cup A_p^H$ 。(4) V_p^i, V_p^O 和 V_p^H 分别是不相交的输入变量集合、输出变量集合和内部变量集合, 记所有变量集合 $V_p = V_p^i \cup V_p^O \cup V_p^H$ 。(5) F_p^V 是一个映射, 把 S_p 中的任何一个状态映射到用 Z 语言描述的状态模式。(6) F_p^A 是一个映射, 把 A_p^I 中的任何一个输入动作映射到用 Z 语言描述的输入操作模式, 把 A_p^O 中的任何一个输出动作映射到用 Z 语言描述的输出操作模式, 把 A_p^H 中的任何一个内部动作映射到用 Z 语言描述的内部操作模式。(7) X 为时钟变量的非负实数有限集合, $C(X)$ 为 X 上时钟约束的集合, 其语法定义为: $\Phi ::= x < c \mid c < x \mid \phi_1 \wedge \phi_2$, 其中 $x \in X, < \in \{<, \leq\}$, c 为非负有理数。(8) 映射 $I: S_p \rightarrow C(X)$ 为每个状态赋予 1 个时间约束, 此约束称为结点不变量。(9) 概率函数 $prob: (S_p \times C(X) \times S_p) \rightarrow [0, 1]$ 为每一条边赋予一实数值概率且对任意一个给定的状态 s 有 $\sum_{(s, a, \varphi, p, \lambda, s') \in T_p} prob(s, s') = 1$, 其中 s' 是 s 的后继状态。(10) TP 是状态之间转换关系的集合, 如果 $(s, a, \varphi, p, \lambda, s') \in T_p, (s, a, \varphi, p) \in prob$, 那么表示在满足转换约束条件 φ 的前提下, 通过动作 $a \in A_p(s)$, 状态 s 以概率 p 可以迁移到新的状态 s' , 同时 $\lambda \subseteq X$ 中的时钟被重置为 0。并且, $((F_p^V(s) \wedge F_p^A(a)) \setminus (x_1, \dots, x_m) \Leftrightarrow F_p^V(t) [y_1'/y_1, \dots, y_n'/y_n])$, 其中 $\{x_1, \dots, x_m\}$ 是 $F_p^V(s)$ 上的变量集合, $\{y_1, \dots, y_n\}$ 是 $F_p^V(t)$ 上的变量集合, $F_p^A(a)$ 上的变量集合是 $\{x_1, \dots, x_m\} \cup \{y_1', \dots, y_n'\}$ 的子集。

2.2 CT-PZIA 对应的时序逻辑 CTML

传统上对概率时间系统的性质描述都是利用概率计算树逻辑 PTCTL。PTCTL^[10] 是对时序逻辑 CTL 的扩展, 在 CTL 基础上加入概率算子 $P \sim p\psi$ 和时间算子 J , 其中 $p \in [0, 1], \sim \in \{<, >, \leq, \geq\}$ 。PTCTL 的语法如下:

$\varphi ::= \text{true} \mid a \mid \varphi \wedge \varphi \mid \neg \varphi \mid P \sim p[\varphi \cup J\psi]$ 。

其中, a 是原子命题, $p \in [0, 1], \sim \in \{<, >, \leq, \geq\}, t$ 为实数。

由于 PTCTL 只局限于反映系统的时序性质, 因此本文

介绍一种可以表示系统度量性质的计算树度量语言^[11] (Computer Tree Measurement Language, CTML)。

与 CTL 和 PTCTL 类似, CTML 也基于状态公式和路径公式, 但是 CTML 公式是用来表达实数值的, 而不仅仅是逻辑真假。在给出 CTML 定义之前, 先给出 CT-PZIA 计算路径的定义。

由于 PTCTL 只局限于反映系统的时序性质, 并不能表达实数值的性质, 例如我们想知道“飞机从开始起飞到结束飞行过程中的平均高度是多少?”, 这样的性质显然不能用 PTCTL 公式来表示, 因此本文提出一种实数值度量语言 CTML。CTML 使用路径操作算子 X 和 U 对应 PCTL 逻辑中的操作算子 $X(\text{Next})$ 和 $U(\text{Until})$ 。CTML 中的“ U ”操作算子被用来做加法或乘法运算直到条件满足, 因此每一个“ U ”算子都标有一个算术操作 $\odot \in \{+, \times\}$ 。与 PCTL 一样, CTML 中的路径算子不能任意嵌套, 必须出现在路径量词后面。CTML 公式中允许值的取值范围依靠操作符, 将状态公式定义为一个函数 $\phi: S \rightarrow R^+$, 约束的状态公式定义为函数 $\varphi: S \rightarrow [0, 1]$, 路径公式定义为 $\psi: S \rightarrow R^+$, 约束的路径公式定义为 $\rho: S \rightarrow [0, 1]$ 。记 AF 为原子状态公式集合, AR 为受限的原子状态集合。在给出 CTML 逻辑公式之前, 有一点需要说明: 由于 PZIA 规范可以对数据变量进行约束, CT-PZIA 中刻画性质不仅有命题逻辑公式, 还有一阶逻辑公式和变量, 仅使用 $\neg, \wedge, \vee, E, U$ 等算子并不能够对数据进行约束, 因此又引入了全称量词 $\forall$, 来描述在一定取值范围内变量的一些性质。

2.2.1 语法

本小节将给出 CTML 的语法定义。

定义 4 $\phi ::= f \mid \phi_1 \sim \phi_2 \mid \phi_1 \odot \phi_2 \mid (\forall x: T) \phi_1 \mid EX\phi_1 \mid E\phi_1 U_{\odot}^t \phi_2$ 。

其中, $\sim \in \{\leq, \geq, <, >\}$, $\odot \in \{+, \times\}$ 。 x 是变量, T 是 x 的类型。

$EX\phi_1$ 表示存在一条路径, 在这条路径上的下一个状态中, ϕ_1 成立。

$E\phi_1 U_{\odot}^t \phi_2$ 表示存在一条路径, 在这条路径上存在一个 $j \leq t$, 且 $\phi_1(P, s_j) > 0$, 对于 s_j 之前的任意状态 s_i 都有 $f(P, s_i) = 0$ 。

2.2.2 语义

在给出 CTML 的语义定义之前, 先给出 CTML 上计算路径的定义。

定义 5 (计算路径) 一个基于连续时间的 CT-PZIA P 中的计算路径有两种无限的状态序列, 一种是离散迁移 $p = (s_0, s_1, \dots)$, 在状态 s_i 上有转移 $(s_i, a, \varphi, pr, \lambda, s_{i+1}) \in TP$, 其中每一个 $i \in \mathbb{N}$; 另一种是推迟迁移。

本文旨在求出路径公式 ψ 的期望值, 因此首先定义一些 ψ 上的属性。 P 上的一个前缀是指一个有限序列 $p = (s_0, \dots, s_{n-1}) \in S^n$, 或者一个无限序列 $p = (s_0, s_1, \dots) \in S^\omega$, 其中 $|p| = n \in \mathbb{N} \cup \{\omega\}$ 是序列的长度。 对于一个给定的前缀 $p = (s_0, \dots, s_{n-1}) \in S_p^n$, S_p^ω 记做以 p 为前缀的所有无限长度路径的集合。 如果 $|p| = p \in \mathbb{N}$, 则有 $S_p^\omega = \{s_0\} \times \dots \times \{s_{n-1}\} \times S^\omega$; 否则, 如果 $|p| = \omega$, 则有 $S_p^\omega = \{p\}$ 。 记 $S_c^\omega = S^\omega$, 其中 $()$ 表示序列长度为 0。

如果对任意路径 $x, x' \in S_p^w, \psi(x) = \psi(x')$, 那么记做前缀 p 决定 ψ 。因为 ψ 在所有的路径上必须有相同的值, 所以数量记做 $\psi(p)$ 。如果对每一条路径 $x \in S^w, \psi(x) > 0$, 则存在一个有限前缀 $p, p \in S_p^w$, 决定 ψ , 记路径公式 ψ 是有限可测的。最后, 对任意有限可测公式 ψ , 定义度量 μ_ψ :

(1) 若 $p = (s_0, s_1, \dots)$ 决定 ψ , 那么 $\mu_\psi(S_p^w) = \psi(p) \prod_{i=1}^{|p|-1} \text{prob}(s_{i-1}, s_i)$;

(2) 否则 $\mu_\psi(S_p^w) = \sum_{s \in S} \mu_\psi(S_{(p,s)}^w)$ 。

其中, $\mu_\psi(\emptyset) = 0$, 因为有限可测性质避免了考察无限长度的路径决定 ψ , 所以对于无限长度路径 x , 有 $\mu_\psi(S_x^w) = 0$ 。下面给出 CTML 的语义定义来求解一个给定公式的值。

定义 6(CTML 的语义) 已知 f 和 g 为原子状态公式, P 为 CT-PZIA 模型, $s \in SP$ 为 P 上的状态, CTML 语义定义如下:

(1) 如果 $h = f$, 那么 $h(P, s) = f(P, s)$, $f(P, s)$ 是一个值的集合;

(2) 如果 $h = f \sim g$, 那么如果 $f(P, s) \sim g(P, s)$ 成立, 则 $h(P, s) = 1$, 否则 $h(P, s) = 0$;

(3) 如果 $h = f + g$, 那么 $h(P, s) = f(P, s) + g(P, s)$, 因为 $f(P, s)$ 和 $g(P, s)$ 都是一个值的集合, 那么利用枚举法可知它们的和也是一个值的集合;

(4) 如果 $h = f \times g$, 那么 $h(P, s) = f(P, s) \cdot (g(P, s))$, 因为 $f(P, s)$ 和 $g(P, s)$ 都是一个值的集合, 所以利用枚举法可知它们的乘积也是一个值的集合;

(5) 如果 $h = (\forall x: T) f$, 那么 $h(P, s) = \bigcap_{v \in T} f\{v/x\}$;

(6) 如果 $h = E\psi$, 那么 $h(P, s) = \mu_\psi(S_s^w)$, 该要求 ψ 在 PZIA 上有限可测;

(7) 如果 $\psi = Xf$, 则 $\psi(s_0, s_1, \dots) = f(P, s_1)$, 公式 Xf 由长度为 2 或者更大的长度的任何前缀决定, 因此有限可测;

(8) 如果 $\psi = f U_{\leq t}^{\leq t} g$, 那么对路径 $p = (s_0, s_1, \dots)$, 若 $j \leq t$, $g(P, s_j) > 0$, 并且对于 s_j 之前的任意状态 s_i 都有 $f(P, s_i) = 0$, 那么 $\psi(p) = \odot_{i=0}^{j-1} f(P, s_i) \cdot g(P, s_j)$; 否则, $\psi(p) = 0$ 。

记 $f U_{\leq \infty}^{\leq \infty} g \equiv f U_{\leq \infty} g$, 路径公式是有限可测的, 考虑任意一条路径 $p = (s_0, s_1, \dots)$, $\psi(p) > 0$ 。对于这条路径, 存在 j 满足第一个条件: $j \leq t, g(P, s_j) > 0$ 并且对于 s_j 之前的任意状态 s_i 都有 $f(P, s_i) = 0$ 。因为公式 $f U_{\leq t}^{\leq t} g$ 由有限前缀 $(s_0, \dots, s_j)$ 决定, 所以它是有限可测的。

状态公式 ϕ 在 PZIA 上的值由 $P_\phi = \sum_{s \in S_P} \phi(P, s) \text{prob}(s)$ 给出。

任何一个 PCTL 公式都可以在 CTML 中表示出来, ϕ' 是 CTML 中一个受限的状态公式, 与 PCTL 状态公式 ϕ 对应。 f' 和 g' 是 CTML 中对应 PCTL 中的状态公式 f 和 g , P 是 PZIA 规范模型, s 是 P 中的一个状态。

(1) (P, s) 满足 ϕ 当且仅当 $\phi'(P, s) = 1$;

(2) (P, s) 不满足 ϕ 当且仅当 $\phi'(P, s) = 0$;

(3) (P, s) 满足 $\neg f$ 等价于 $\phi'(P, s) = f'(P, s) < \text{one}$;

(4) (P, s) 满足 $f \wedge g$ 等价于 $\phi'(P, s) = f'(P, s) \cdot g'(P, s)$;

(5) (P, s) 满足 $f \vee g$ 等价于 $\phi'(P, s) = (f'(P, s) + g'(P, s)) > \text{zero}$;

(6) (P, s) 满足 $f U \leq t g$ 等价于 $\phi'(P, s) = f'(P, s) U_{\leq t}^{\leq t} g'(P, s)$;

(7) (P, s) 满足 $P \sim v \psi$ 等价于 $\phi'(P, s) = E\psi' \sim v$, 其中 $\sim \in \{\leq, \geq, <, >\}$, CTML 中的 ψ' 等价于 PCTL 中的路径公式 ψ 。

3 等价域构造方法

对于变迁系统 $S(P)$ 而言, 状态空间 $S \times V(X)$ 集合是无限的, 而现在的模型检测技术只能处理有穷状态系统, 因此必须把无穷状态转化为有穷状态^[15]。为了获得 $S(P)$ 的无穷状态空间的有穷表示, Alur, Courcoubetis 和 Dill^[16-17] 提出一种时钟等价方法, 把时间自动机等价于域自动机, 但是按照 Alur 的时钟等价方法构造出的域自动机存在状态空间迅速膨胀爆炸的问题。本文采用一种优化的时钟等价方法, 并且为了满足优化的时钟等价方法, 修改了域自动机定义, 使等价后的域自动机状态数尽量少^[18]。这样可以将 CT-ZIA 等价作为一种带 Z 的域自动机, 从而使 CT-ZIA 的模型检测问题成为可能。

定义 7 设 c 为时钟的约束常量值。

对时钟 x 而言, c_x 表示时钟 x 的约束常量值。若从 s 结点到 s' 结点的边上有一时钟约束 $x \geq 2$, 即 $v(x) \geq c_x$ 时, 变迁转换允许发生, 则 $c_x = 2$ 。

定义 8 域 r 是状态结点和时钟区域的结合, 可以描述为一个二元对 (s, v_p) , 其中, $s \in S, v_p$ 为时钟关键点, 且 $v_p \in V(X)$ 。 $s_{\approx}$ 是 $s = (s, v)$ 对应于 (s, v_p) 的缩写。

定义 9 设 r 是一个无界域, 表示对所有的时钟 $v(x) \geq c_x, x \in X$ 。

定义 10 设 r 的后继域为 $\text{succ}(r)$, r 是一个无界域, 则 $r = \text{succ}(r)$; r 和 r' 为两个不同的域, 即 $r \neq r', r = \text{succ}(r)$ 表示存在正实数 $d_p \in \mathbb{R}^+$, 即 $d_p: v_{p_2} - v_{p_1}, v_{p_2} > v_{p_1}$ 使 $r = s_{\approx}, s \xrightarrow{d_p} s', r = (s + d_p)_{\approx}$ 成立。

定义 11 域自动机 $R(P, \phi)$ 是一个转换系统 $(R, r, \rightarrow)$, 定义如下:

$R = S_{/\approx} = \{s_{\approx} \mid s \in S\}$;

$r_0 = s_{0_{\approx}}$;

$r \rightarrow r'$ 当且仅当 $\exists s, s' r = s_{\approx} \wedge r' = s'_{\approx} \xrightarrow{*} s'$;

$r \rightarrow r'$ 当且仅当 $r' = \text{succ}(r)$ 。

对状态 s 来说, 约束条件 $x \geq 2$ 中的约束常量 2 是一个关键点, 因为当 $v(x) > 2$ 或 $v(x) < 2$ 时, 无论取何值都是不相关的, 所以仅需考虑约束常量这个关键点。综上所述, 对时钟 x 而言, 可分为 3 种情况:

1) $v(x) < c_x$ 则 $v'(x) < c_x$;

2) $v(x) = c_x$ 则 $v'(x) = c_x$;

3) $v(x) > c_x$ 则 $v'(x) > c_x$ 。

考虑时钟存在复位操作, 0 也是一个特定的关键点。如果一个时钟有 n 个不同约束常量, 那么就存在 $n+1$ 个时钟关键点。假定时钟关键点集合 C_x 有 $c_{x_0} < c_{x_1} < \dots < c_{x_k}$ 成立, 且 $c_{x_0} = 0, c_{x_i} \in C_x$, 其中 $i = 0, 1, \dots, k$, 则优化时钟等价方法(1)如下:

- 1) $c_{x_{i-1}} < v(x) < c_{x_i}$, 则 $c_{x_{i-1}} < v'(x) < c_{x_i}$;
- 2) $v(x) > c_{x_k}$, 则 $v'(x) > c_{x_k}$;
- 3) $v(x) = c_{x_k}$, 则 $v'(x) = c_{x_k}$ 。

通过以上等价, 时钟就变为有限集合。

如果存在两个时钟 x, y , 以及相应时钟关键点集合 C_x 和 C_y , 则可以用一个二元对 (c_x, c_y) 来描述时钟关键点, 且 $(c_x, c_y) \in C_x \times C_y$ 。然而时钟等价不仅需要考虑每个时钟的取值范围, 而且还需要考虑 $y-x$ 的取值范围。为了确定 $y-x$ 的取值范围, 首先要确定 $y-x$ 的分界线, 根据时钟关键点来定义分界线 $y-x=h$, 其中 $h=c_y-c_x, c_y \in C_y$ 且 $0 < y < c_y, c_x \in C_x$, 且 $0 < x < c_x$ 。假定两个时钟关键点集合 C_x 和 $C_y, c_{x_i} \in C_x, c_{y_i} \in C_y$, 并满足 $c_{x_0} < c_{x_1} < \dots < c_{x_k}, c_{y_0} < c_{y_1} < \dots < c_{y_l}$, 且 $h_1 < h_2 < h_m, m=k \times l$ 。那么时钟等价方法(2)如下:

- 1) 如果 $h_{s-1} < v(y) - v(x) < h_s$, 且 $c_{x_{i-1}} < v(x) < c_{x_i}, c_{y_{j-1}} < v(y) < c_{y_j}$, 则 $h_{s-1} < v'(y) - v'(x) < h_s$, 且 $c_{x_{i-1}} < v'(x) < c_{x_i}, c_{y_{j-1}} < v'(y) < c_{y_j}$, 其中 $c_{y_{j-1}} - c_{x_i} < h_{s-1}, h_s < c_{y_j} - c_{x_{i-1}}, i=1, \dots, k, j=1, \dots, l, s=2, \dots, m$;
- 2) 如果 $v(x) - v(y) = h_s$, 且 $c_{x_{i-1}} < v(x) < c_{x_i}, c_{y_{j-1}} < v(y) < c_{y_j}$, 则 $v'(x) - v'(y) = h_s$ 且 $c_{x_{i-1}} < v(x) < c_{x_i}, c_{y_{j-1}} < v'(y) < c_{y_j}$, 其中 $c_{y_{j-1}} - c_{x_i} < h_s < c_{y_j} - c_{x_{i-1}}, i=1, \dots, k, j=1, \dots, l, s=1, \dots, m$ 。

当考虑两个时钟时, 还可以对它进行一些限制, 因为仅当 $v(x) < c_{x_{\max}}, v(y) < c_{y_{\max}}$ 时, 才需考虑两个时钟之间的大小关系, 否则无需考虑两个时钟之间的大小关系。

4 性质验证

4.1 验证算法

算法 1 CTML 性质验证

对于 CTML 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 以及 SP 中的每一个状态 s , 均已知 ϕ 在状态 s 上的取值范围。

输入: CT-PZIA P ; CTML 公式 $\phi; s \in SP$ 为 P 上的状态

输出: CTML 公式 ϕ 在 CT-PZIA 模型 P 中的状态 s 上的值的取值范围
过程:

Case 1 当 $\phi = f$ 时, $\phi(P, s) = f(P, s)$ 。

Case 2 当 $\phi = \phi_1 \sim \phi_2$ 时,

(1) 若 $\phi_1(P, s) \sim \phi_2(P, s)$, 则 $\phi(P, s) = 1$;

(2) 否则, $\phi(P, s) = 0$ 。

Case 3 当 $\phi = \phi_1 + \phi_2$ 时, $\phi(P, s) = \phi_1(P, s) + \phi_2(P, s)$ 。

Case 4 当 $\phi = \phi_1 \times \phi_2$ 时, $\phi(P, s) = \phi_1(P, s) \cdot \phi_2(P, s)$ 。

Case 5 当 $\phi = (\forall x: T) \phi_1$ 时, $\phi(P, s) = \bigcap v \in T \phi_1\{v/x\}$ 。

Case 6 当 $\phi = EX \phi_1$ 时,

$$\phi(P, s) = \sum_{s_1 \in S_P} f(P, s_1) \text{prob}(s, s_1)$$

Case 7 当 $\phi = E \phi_1 U^{\leq t} \phi_2$ 时,

$$\begin{aligned} \phi_t(P, s_0) &= \sum_{(s_1, \dots, s_t) \in S^t} (f(P, s_0) + \dots + f(P, s_{t-1})) \prod_{i=1}^t \text{prob}(s_{i-1}, s_i) g(P, s_t) \\ &= \sum_{(s_1, \dots, s_t) \in S^t} f(P, s_0) \prod_{i=1}^t \text{prob}(s_{i-1}, s_i) g(P, s_t) + \sum_{s_1 \in S} \text{prob}(s_0, s_1) h_{t-1}(P, s_1) \end{aligned}$$

循环终止条件为: 对所有的状态 $s, \phi_0(P, s) = 0$ 。定义 $\phi_t(P, s_0)$ 为“PZIA 模型从状态 s_0 开始在 t 时刻到达状态 s_t 并且 $g(P, s_t) > 0$ 时, 根据 f 累积的奖励数量”。

Case 8 当 $\phi = E \phi_1 U^{\leq t} \phi_2$ 时,

$$\begin{aligned} \phi_t(P, s_0) &= \sum_{(s_1, \dots, s_t) \in S^t} f(P, s_0) \dots f(P, s_{t-1}) \prod_{i=1}^t \text{prob}(s_{i-1}, s_i) g(P, s_t) \\ &= \sum_{(s_1, \dots, s_t) \in S^t} f(P, s_0) \text{prob}(s_0, s_1) h_{t-1}(P, s_1) \end{aligned}$$

循环终止条件为 $\phi_0(P, s) = g(P, s)$ 。

4.2 算法正确性证明

命题 1 令 ϕ 为 CTML 公式, 假设 ϕ 的所有子公式已经被处理过, 即对于 ϕ 的任意子公式 φ 及 CT-PZIA P 中的每一个状态 s , 均已知 φ 在状态 s 上的取值范围, 计算公式 ϕ 在 CT-PZIA 模型 P 中的状态 s 上的取值范围 $\phi(P, s)$ 。

证明: 下面利用数学归纳法对 ϕ 的结构进行归纳证明:

(1) 若 $\phi = f$, 有 $\phi(P, s) = f(P, s)$;

(2) 若 $\phi = \phi_1 \sim \phi_2$, 根据归纳假设已知 $\phi_1(P, s)$ 和 $\phi_2(P, s)$ 的取值范围, 那么如果 $\phi_1(P, s) \sim \phi_2(P, s)$, 则有 $\phi(P, s) = 1$; 否则有 $\phi(P, s) = 0$;

(3) $\phi = (\forall x: T) \phi_1$, 根据归纳假设已知 $\phi_1(P, s)$ 的取值范围, 那么 $\phi(P, s) = \bigcap v \in T \phi_1\{v/x\}$;

(4) 若 $\phi = \phi_1 + \phi_2$, 根据归纳假设已知 $\phi_1(P, s)$ 和 $\phi_2(P, s)$ 的取值范围, 那么可知 $\phi_1(P, s) + \phi_2(P, s)$, 也就可求出 $\phi(P, s) = \phi_1(P, s) + \phi_2(P, s)$ 的取值范围;

(5) 若 $\phi = \phi_1 \times \phi_2$, 根据归纳假设已知 $\phi_1(P, s)$ 和 $\phi_2(P, s)$ 的取值范围, 那么可知 $\phi_1(P, s) \cdot \phi_2(P, s)$, 也就可求出 $\phi(P, s) = \phi_1(P, s) \cdot \phi_2(P, s)$ 的取值范围;

(6) 若 $\phi = EX \phi_1$, 根据归纳假设已知 $\phi_1(P, s)$ 的取值范围, 即在 PZIA 模型中存在一条路径上的下一个状态上 ϕ_1 的取值范围为 $\phi_1(P, s)$, 那么 $\psi(s_0, s_1, \dots) = f(P, s_1)$, 进而可知 $\phi(P, s) = \sum_{s_1 \in S_P} f(P, s_1) \text{prob}(s, s_1)$;

(7) 若 $\phi = E \phi_1 U^{\leq t} \phi_2$, 根据归纳假设已知 $\phi_1(P, s)$ 和 $\phi_2(P, s)$ 的取值范围, 在 PZIA 模型 P 中存在一条路径上的状态 $s_j, j \leq t, g(P, s_j) > 0$, 并且对于 s_j 之前的任意状态 s_i 都有 $f(P, s_i) = 0$, 那么 $\psi(p) = \sum_{i=0}^{j-1} f(P, s_i) \cdot g(P, s_j)$, 可知 $\phi(P, s) = \mu_\psi(S_s^w)$;

(8) $\phi = E \phi_1 U^{\leq t} \phi_2$, 根据归纳假设已知 $\phi_1(P, s)$ 和 $\phi_2(P, s)$ 的取值范围, 在 PZIA 模型 P 中存在一条路径上的状态 $s_j, j \leq t, g(P, s_j) > 0$, 并且对于 s_j 之前的任意状态 s_i 都有 $g(P, s_i) = 0$, 那么 $\psi(p) = \prod_{i=0}^{j-1} f(P, s_i) \cdot g(P, s_j)$, 可知 $\phi(P, s) = \mu_\psi(S_s^w)$ 。

5 模型检测算法在现有工具基础上的实现问题

现有的常见模型检测工具大多针对离散时间, 也有少部分工具针对连续时间, 但不能直接用这些工具来实现本文算法, 原因如下:

1) 本文所使用的模型 CT-PZIA 是带有数据变量约束的,

(下转第 593 页)

数据子集。整体准确率结果如图 3 所示。

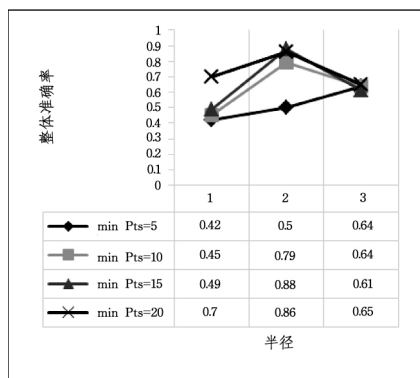


图 3 算法整体准确率

在 ϵ 为 0.02 且 $\min Pts$ 为 15 时,实验的整体准确率取得最大值,到达 88%,取得了理想的效果,此时算法的精确率为 0.91,算法的聚类时间平均为 120s。同时,通过实验可知,本文所用算法对输入参数 ϵ , $\min Pts$ 较为敏感。

(上接第 574 页)

现有常见工具(如 UPPAAL)所基于的模型(如时间自动机)上没有数据约束;

2)本文的模型中包含的变量既包括连续变量也包括离散变量,现有常见的工具所基于的模型通常只包含离散变量。

如果要在现有工具的基础上实现本文算法,需要把本文所用的模型转换为现有工具所基于的模型,但存在以下两个方面的问题:

1)将本文的模型转换为现有工具所基于的模型会导致转换的模型比较大,例如,一个状态 $0 < x < 3$ 就变成 $x=1, x=2$ 两个状态;状态 $x > 3$ 就变成 $x=4, x=5, \dots$ 等多个状态,这样会导致转换的模型状态非常多,使得模型检测的复杂度增加。

2)因为本文的逻辑公式 CTML 与现有工具所基于的模型对应的时序逻辑公式(如 PTCTL)不同,所以也要将模型对应的时序逻辑公式进行转换,同理会导致逻辑公式长度变大,使得模型检测的复杂度增加。

结束语 为了对带有数据约束的概率实时系统进行研究,提出了一种基于连续时间的 CT-PZIA 规范,其既可以描述系统的行为属性,也可以刻画系统数据方面的属性;并给出了基于连续时间 CTML 的模型检测算法,从而实现了带数据约束的概率实时系统的模型检测。本文研究了带数据约束的概率实时系统的原理和方法,下一步的研究工作考虑将这些原理和方法运用到具体的实际概率实时系统中,这是未来研究工作的方向和重点。

参考文献

- [1] 李广元,唐稚松.带有时钟变量的线性时序逻辑与概率实时系统验证[J].软件学报,2002,13(1):33-41.
- [2] 戎玫,张广泉.模型检测新技术研究[J].计算机科学,2003,30(5):102-104.
- [3] BÉRARD B, BIDOIT M, FINKEL A, et al. Systems and software verification: model-checking techniques and tools [M]. Springer Publishing Company, Incorporated, 2010.
- [4] 王晶,戎玫,张广泉,等.基于概率模型检测的 Web 服务组合验

结束语 本文运用 DBSCAN 算法对电子邮件网络进行社团发现。在介绍 DBSCAN 算法的基础上,详细设计了算法在邮件网络社团发现中的具体实现流程,最后通过对安然邮件语料集的实验证明了算法的有效性以及在社团发现方面的优越性。

参考文献

- [1] 杨博,刘大友,金第,等.复杂网络聚类方法[J].软件学报,2009,20(1):54-66.
- [2] 罗浪,张绍武,陈韬.基于稠密子团和边聚类系数的局部社团挖掘算法[J].电子设计工程,2013,21(18):36-40.
- [3] NEWMAN M E J. Detecting community structure in networks [J]. European Physical Journal, 2004, 38(2): 321-330.
- [4] 林旺群,卢风顺,丁兆云,等.基于带权图的层次化社团并行计算方法[J].软件学报,2012,23(6):1517-1530.
- [5] 马菲,徐汀荣.基于种子边的重叠社团发现算法[J].计算机应用研究,2015,32(9):2563-2596.
- [6] 证[J]. 计算机科学,2012,39(1):120-123.
- [5] CAO Z, WANG H. Extending interface automata with Σ notation [M]// Fundamentals of Software Engineering. Springer Berlin Heidelberg, 2012: 359-367.
- [6] CAO Z, WANG H. Hybrid ZIA and its Approximated Refinement Relation [C]// Proceedings of the 6th International Conference on Evaluation of Novel Approaches to Software Engineering. Beijing, China, 2011: 260-265.
- [7] 狄杨思.形式规范的自动验证算法的研究[D].南京:南京航空航天大学,2012.
- [8] 狄杨思,曹子宁,王辉.一种基于形式规范 ZIA 的自动验证方法[J].光电与控制,2011,18(增刊):113-120.
- [9] CLARKE E M, GRUMBERG O, PELED D A. Model checking [M]. MIT Press, 1999.
- [10] 周从华,刘志锋,王昌达.概率计算树逻辑的限界模型检测[J].软件学报,2012,23(7):1656-1668.
- [11] MINER A S, JING Y. A formal language toward the unification of model checking and performance evaluation [M]// Analytical and Stochastic Modeling Techniques and Applications. Springer Berlin Heidelberg, 2010: 130-144.
- [12] KOYMANS R. Specifying message passing and time-critical systems with temporal logic [M]. Springer, 1992.
- [13] EMERSON E A, MOR A K, SISTLA A P, et al. Quantitative temporal reasoning [M]// Computer-Aided Verification. Springer Berlin Heidelberg, 1991: 136-145.
- [14] ALUR R, DILL D L. A theory of timed automata [J]. Theoretical computer science, 1994, 126(2): 183-235.
- [15] ALUR R. Timed automata [M]// Computer Aided Verification. Springer Berlin Heidelberg, 1999: 8-22.
- [16] ALUR R. Techniques for automatic verification of real-time systems [D]. Stanford University, 1991.
- [17] ALUR R, COURCOUBETIS C, DILL D. Model-checking for real-time systems [C]// Fifth Annual IEEE Symposium on Logic in Computer Science, 1990 (LICS'90). IEEE, 1990: 414-425.
- [18] 钱俊彦,赵岭忠,古天龙.一种基于时间自动机的时钟等价性优化方法[J].计算机工程,2005,31(18):71-73.